NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

EFFICIENT EMPLOYMENT OF IMPERFECT SEARCH SENSORS IN COMPLEX ENVIRONMENTS

by

Kurt E. Wilson

September 2009

Thesis Advisor:	Roberto Szechtman
Second Reader:	Johannes O. Royset

Approved for public release; distribution is unlimited

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704-0188*
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.			
1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** September 2009	**3. REPORT TYPE AND DATES COVERED** Master's Thesis	
4. TITLE AND SUBTITLE Efficient Employment of Imperfect Search Sensors in Complex Environments			**5. FUNDING NUMBERS**
6. AUTHOR(S) Kurt E. Wilson			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School, Monterey, CA 93943-5000			**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			**12b. DISTRIBUTION CODE** A

13. ABSTRACT (maximum 200 words)

Defense planners must strive to develop and incorporate new, efficient procedures to allocate scarce resources in varied complex environments. We consider two discrete-time, discrete-space search effort allocation situations. Both involve the employment of an imperfect sensor, which is subject to both false-positive and false-negative errors. The area of interest, comprised of several disjoint area-cells, contains a single target of interest. In the first situation, the target moves according to a Markovian transition matrix, which is unknown to the sensor operator. The objective is to estimate the target's steady-state distribution, using only the sensor's detection signals and knowledge of its false-positive and false-negative rates. The second situation considers a stationary target, wherein the objective is to determine the area-cell occupied by the target, in the fewest expected number of investigations, to within certain operator-prescribed error tolerances. We develop an *adaptive* algorithm based on *stochastic approximation* for the first situation, and show that the resultant rate of error in determining target presence/absence in any area-cell converges to zero at the fastest possible rate. We propose a *sequential elimination procedure* for the second situation, which provides an efficient determination of target location and guarantees its error rate not to exceed the operator-prescribed tolerance.

14. SUBJECT TERMS Search Theory, Stochastic Search and Optimization, Stochastic Approximation, Sequential Analysis, Markov Motion			**15. NUMBER OF PAGES** 97
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

EFFICIENT EMPLOYMENT OF IMPERFECT SEARCH SENSORS IN COMPLEX ENVIRONMENTS

Kurt E. Wilson
Lieutenant Commander, United States Navy
B.A., Iowa State University, 1996

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 2009

Author: Kurt E. Wilson

Approved by: Roberto Szechtman
Thesis Advisor

Johannes O. Royset
Second Reader

Robert F. Dell
Chairman, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Defense planners must strive to develop and incorporate new, efficient procedures to allocate scarce resources in varied complex environments. We consider two discrete-time, discrete-space search effort allocation situations. Both involve the employment of an imperfect sensor, which is subject to both false-positive and false-negative errors. The area of interest, comprised of several disjoint area-cells, contains a single target of interest. In the first situation, the target moves according to a Markovian transition matrix, which is unknown to the sensor operator. The objective is to estimate the target's steady-state distribution, using only the sensor's detection signals and knowledge of its false-positive and false-negative rates. The second situation considers a stationary target, wherein the objective is to determine the area-cell occupied by the target, in the fewest expected number of investigations, to within certain operator-prescribed error tolerances. We develop an *adaptive* algorithm based on *stochastic approximation* for the first situation, and show that the resultant rate of error in determining target presence/absence in any area-cell converges to zero at the fastest possible rate. We propose a *sequential elimination procedure* for the second situation, which provides an efficient determination of target location and guarantees its error rate not to exceed the operator-prescribed tolerance.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I. INTRODUCTION 1
- A. MODEL COMMONALITIES 1
- B. SINGLE MARKOV TARGET (*SMT*) MODEL 2
  - 1. Problem Statement 2
  - 2. Objective 2
  - 3. Scope and Limitations 3
- C. SINGLE STATIC TARGET (*SST*) MODEL 3
  - 1. Problem Statement 3
  - 2. Objective 4
  - 3. Scope and Limitations 4

II. LITERATURE REVIEW 7
- A. BACKGROUND 7
  - 1. Sensors and Complex Environments 7
  - 2. Search Theory and Mission Planning 9
- B. STOCHASTIC SEARCH AND OPTIMIZATION 14
- C. SEQUENTIAL ANALYSIS 16
- D. LITERATURE REVIEW CONCLUSIONS 18

III. MODEL DEVELOPMENT 21
- A. *SMT* MODEL 21
  - 1. Basic Framework 21
  - 2. Large Search Budget Results 22
  - 3. Stochastic Approximation Algorithm 24
- B. *SST* MODEL 26
  - 1. Case 1: Independent Area-cells with No Target or One Target 26
  - 2. Case 2: Single Target and Two Area-cells 29
  - 3. Case 3: Single Target in One of m>2 Area-cells 32

IV. COMPUTATIONAL STUDY 35
- A. *SMT* MODEL 35
  - 1. Scenario Development 35
  - 2. Stochastic Approximation Algorithm Implementation and Results 41
  - 3. Analysis of Stochastic Approximation Algorithm Performance 44
- B. *SST* MODEL 47
  - 1. Scenario Development 47
  - 2. Sequential Eliminating Procedure Implementation and Results 48
  - 3. Analysis of Sequential Eliminating Procedure Performance 50

V. CONCLUSIONS AND FUTURE WORK 55
- A. CONCLUSIONS 55
  - 1. Single Markov Target Model 55
  - 2. Single Static Target Model 55

B. AREAS FOR POSSIBLE FUTURE STUDY 56
1. Single Markov Target Model 56
2. Single Static Target Model 56

APPENDIX A 59

APPENDIX B 63

APPENDIX C 67

LIST OF REFERENCES 71

INITIAL DISTRIBUTION LIST 75

LIST OF FIGURES

Figure 1. Example *Threat Mapper* input (From Riese, 2008).......................................13
Figure 2. Example *Threat Mapper* output (From Riese, 2008).....................................14
Figure 3. Example sample paths of the one-cell sequential procedure. 29
Figure 4. Example sample paths for the two-cell *SST* model case.................................32
Figure 5. Discretized notional scenario Area of Operational Interest. 36
Figure 6. Number of looks versus steady-state estimate for one sample path of three particular area-cells. 44
Figure 7. Large number of looks versus the logarithm of probability of error. 46
Figure 8. Small number of looks versus the logarithm of probability of error. 47
Figure 9. Polynomial increase in number of looks as difference between a and b gets small $(m=1,\ \alpha=0.05)$. 48
Figure 10. Near-linear relationship between m and number of looks, case 3, $\alpha=0.05$. 49
Figure 11. Effect of varying α on expected number of looks (m=10). 50
Figure 12. Effect of varying α on observed error rates (m=10).......................................52
Figure 13. Achieved error rates with a threshold of 0.05, based on 5,000 iterations at each integer level of 3 to 156 cells...53

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

Table 1. Initial TOI probability map, derived by normalizing the aggregate *Threat Map* (After Riese, 2008). 38
Table 2. Notional TOI steady-state distribution....................39
Table 3. Notional hyperspectral sensor *sensitivity* (center cell values) and *specificity* (bottom cell values). 40
Table 4. Theoretical optimal search frequencies (center cell values) and *SA* algorithm resultant search frequencies (bottom cell values, based on 50,000 iterations). 42
Table 5. TOI transition matrix (non-zero columns only). (Sheet 1 of 3). 67
Table 6. TOI transition matrix (non-zero columns only). (Sheet 2 of 3). 68
Table 7. TOI transition matrix (non-zero columns only). (Sheet 3 of 3). 69

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

AOI	Area of Operational Interest
CCD	Coherent Change Detection
CDP	Cumulative Detection Probability
FAB	Forward and Backward
F^2T^2E	Find, Fix, Track, Target, Engage
HITL	Human-in-the-Loop
IED	Improvised Explosive Device
IID	Independent Identically Distributed
ISAR	Inverse Synthetic Aperture Radar
ISR	Intelligence, Surveillance, and Reconnaissance
LOS	Line-of-Sight
MOE	Measure of Effectiveness
MOP	Measure of Performance
PMF	Probability Mass Function
RD	Recognition Differential
SA	Stochastic Approximation
SMT	Single Markov Target
SST	Single Static Target
SAR	Synthetic Aperture Radar
TOI	Target of Interest
TTPs	Tactics, Techniques, and Procedures
UAS	Unmanned Aerial System

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

Today's operational planners and sensor operators are challenged with scarce resources in terms of both time and sensor assets. One may argue that recent advancements in technology have slowed the growth of innovative employment of tactics, techniques, and procedures (TTPs), making sensor operators increasingly more dependent upon that technology. Defense planners must strive to develop and incorporate new, efficient procedures to allocate scarce resources in many different complex environments. Any efficiency that can be gained, however small, may have a compound effect over time on overall combat readiness, by freeing up precious assets to perform other time-sensitive, critical sensing actions.

In this thesis, we consider two particular search effort allocation situations of operational interest. We begin by describing some characteristics common to both situations, then follow with a discussion of both situations' unique properties and a discussion of our proposed models. Both situations allow us to contend with the fact that search sensors are imperfect; i.e., they are subject to declaring a target present when it is in fact absent (false-positive error), as well as to declaring a target absent when it is in fact present (false-negative error). The Area of Operational Interest (AOI) for both situations is comprised of a grid of discrete, non-overlapping area-cells, each cell having its own associated values for sensor error rates. These area-cells might be defined by geo-political borders, terrain features, or some arbitrary grid system of tactical significance to the operator, and need not be uniform in size nor shape. Both situations involve a single target of interest (TOI), located somewhere within the AOI. The TOI is unintelligent, in the sense that it does not react to any sensing action. We treat time in both of these situations in terms of discrete time-steps. The operator makes one *investigation* into one area-cell per time-step. Both situations deal with the *allocation* of search sensors, which implies that we are concerned with the placement of sensors, and that those sensors are not restricted to follow any particular path. In contrast, a search *path* problem might impose such a restriction; say, for example, only immediately adjacent area-cells may be investigated on subsequent time-steps. It may be assumed

either that sufficient time exists between investigations so that a single sensor may be re-positioned to any other area-cell, or that each area-cell contains one pre-positioned sensor, but the operator may process only one of those sensors in any time-step.

The first situation deals with a randomly moving target, whose underlying pattern of motion results in some steady-state distribution *over time*. This suggests that this particular circumstance is concerned with long-term Intelligence, Surveillance and Reconnaissance (ISR) operations, where the number of available search opportunities for the operator is rather substantial. For a simplified illustration of this steady-state concept in a two-cell AOI, this may mean that, in the long-run average, the TOI is present in area-cell "*x*" 25% of the time, and in area-cell "*y*" 75% of the time. Of course, the underlying true steady-state distribution of the TOI's location is ultimately unknown to the sensor operator, who must use only the (imperfect) detection signals provided by the sensors, along with knowledge of their associated error rates, to determine an estimate of this steady-state distribution.

The model developed in this thesis to deal with this first situation is an *adaptive* model, meaning that it provides a dynamic allocation plan based on new information as it becomes available. We show that our particular procedure converges to the true steady-state distribution for a large number of search opportunities, and that the error rate for determining target presence/absence in any cell converges to zero more quickly than with other allocation schemes. The result implies a cost-savings to the sensor operator and the operational planner, allowing precious assets to be freed up to perform other, time-critical sensing evolutions.

In the second search allocation situation of this thesis, we are concerned with a stationary target hidden somewhere in the AOI. Likely candidates fitting this template might include an insurgent in hiding, an Improvised Explosive Device (IED), or a downed friendly aircraft for search and rescue. The objective in this circumstance is to determine, in the smallest expected number of investigations, the area-cell in which the target is located. Of course, since the sensors are imperfect, there is no guarantee that the answer is correct, so in this case the answer must be framed with some sort of confidence. To accomplish this, the operator prescribes an error tolerance. For example, an error tolerance of 5% would mean that the operator is willing to accept that the model provides a correct determination of target presence at least 95% of the time. Naturally, the larger the error tolerance the operator is willing to accept, the more quickly the operator can expect to make a determination. Conversely, a small tolerance for error could mean many more search attempts expended to make a determination.

To handle this second situation, we develop a family of sequential elimination procedures. These procedures work in stages; during each stage, all possible area-cells in contention of hiding the target are examined and ranked. If any area-cell, when compared to the area-cell of maximum likelihood, fails to meet a certain threshold, that area-cell is eliminated permanently from contention. The process continues until only one area-cell remains in the pool of candidates, and that cell is declared to contain the target. We show that our sequential models provide efficient solutions to this class of problem, while guaranteeing to meet the user-prescribed error tolerances. In particular, we show our procedure not only outperforms a typical sensible approach that uses the same expected number of investigations; indeed the sensible approach fails to meet the error tolerance. As with our adaptive model for the first situation, the cost-savings to the sensor operator and operational planner when implementing this sequential eliminating procedure is evident; scarce resources may be more readily available to perform other critical sensing tasks.

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

I would first like to express my greatest love and appreciation of my wife, Jennifer, and sons, Brady and Graham, whose support and understanding during my long hours of travel, research, and writing were significant. I would also like to thank Assistant Professor Roberto Szechtman for agreeing to take on this project of mutual interest, and for the substantial amount of time and effort he put forth to assure my completion of this thesis. Without his gift of immense theoretical and technical insight, the ideas of this thesis would never have progressed beyond the back of an envelope. Additionally, I must gratefully acknowledge Assistant Professor Johannes Royset for providing highly relevant and constructive feedback as second reader. Finally, I would like to thank the extremely professional staff of the Global Engagement Department of the Johns Hopkins University Applied Physics Lab. Their pride in work and professionalism is immediately and continually evident, and they went far beyond the call of duty during my experience tour to provide immeasurable support. I would particularly like to express my appreciation to Jack Keane for putting the pieces in place under short notice to allow me to participate in the experience tour, and to Brian Funk for serving as my sponsor during my six enlightening weeks at the lab.

This thesis is humbly dedicated to those who support us who serve.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

Today's operational planners and sensor operators are challenged with scarce resources in terms of both time and sensor assets. One may argue that recent advancements in technology have slowed the growth of innovative employment tactics, techniques, and procedures (TTPs), making sensor operators increasingly more dependent upon that technology. Defense planners must strive to develop and incorporate new, efficient procedures to allocate scarce resources in many different complex environments. Any efficiency that can be gained, however small, may have a compound effect over time on overall combat readiness, by freeing up precious assets to perform other time-sensitive, critical sensing actions.

In this thesis, we consider two particular search effort allocation situations of operational interest. We refer to these as the Single Markov Target (*SMT*) model, and the Single Static Target (*SST*) model. We begin by describing some characteristics common to both models, then follow with discussions of the background and problem statements, objectives, and scope and limitations for the *SMT* and *SST* models, treated in Section B and Section C, respectively.

A. MODEL COMMONALITIES

Both the *SMT* and *SST* models allow us to contend with the fact that search sensors are imperfect; i.e., they are subject to declaring a target present when it is in fact absent (false-positive error), as well as to declaring a target absent when it is in fact present (false-negative error). The Area of Operational Interest (AOI) for both models is comprised of a grid of discrete, non-overlapping area-cells, each cell having its own associated values for sensor error rates. These area-cells might be defined by geo-political borders, terrain features, or some arbitrary grid system of tactical significance to the operator, and need not be uniform in size nor shape. Both models involve a single target of interest (TOI), located somewhere within the AOI. The TOI is *non-reactive*; i.e., it does change its pattern of behavior in response to any sensing action. We treat time in both of these models in terms of discrete time-steps. The operator makes one *investigation* into one area-cell per time-step. Both models deal with the *allocation* of search sensors, which implies that we are concerned with the placement of sensors, and that those sensors are not restricted to follow any particular path. In contrast, a search *path* problem might impose such a restriction, say, e.g., only immediately adjacent area-cells may be investigated on subsequent time-steps. It may be assumed either that sufficient time exists between investigations so that a single sensor may be re-positioned to any other area-cell, or that each area-cell contains one pre-positioned sensor, but the operator may process only one of those sensors in any time-step.

In summary, both *SMT* and *SST* are discrete-time, discrete space, single non-reactive target, single searcher, path unconstrained, imperfect search sensor allocation models.

B. SINGLE MARKOV TARGET (*SMT*) MODEL

1. Problem Statement

Consider an AOI, in which a TOI is known to be operating. The TOI could be a convoy, a vehicle, or an individual insurgent. Assume that, based on intelligence data and social theory, this particular TOI is subject to movement in a *Markovian* fashion. That is to say, at each time step, the TOI moves randomly according to some probability mass function (pmf), which may depend on a finite number of current and past locations. An area-cell is considered to be determined when enough evidence exists for the sensor operator to declare that a target is either present or absent in that cell. The resultant operational problem is summarized: how to estimate the steady-state distribution of the TOI (whose transition matrix and resultant steady-state distribution clearly are unknown to the searcher) adequately, based solely on noisy observations from imperfect sensors.

2. Objective

The Strong Law of Large Numbers suggests that any search effort scheme that allocates a positive allocation of effort to every possible area-cell leads to estimates that converge upon the true steady state distribution of the target (see Chapter III for a more thorough discussion of this). Here, the objective is to determine an allocation scheme that converges quickly, while displaying an improved error decay rate (where the error is an incorrectly determined area-cell) when compared to other schemes for cases in which the available number of search opportunities—the *search budget*—is large.

3. Scope and Limitations

For the scenario used in this portion of the thesis, we consider only one single target, whose movement is characterized as *Markovian.* A target that moves according to some other scheme would not be appropriate for this model. We do not consider an intelligent or reactive target. Additionally, we consider only a single sensor (or multiple sensors subject to the constraint that only one sensor may be used at a particular time-step); therefore, we do not consider cooperation among sensors. It is assumed that sensor sensitivity and specificity (see Chapter II for the associated definitions) for each area are *known values*. In reality, it is likely that these values would be noisy; the manufacturer might provide to the operator their expected values as published specifications, or perhaps the operator might derive them using some form of tactical decision aid. Our model does not take into account that the sensitivity and specificity values might be correlated with the number of looks; e.g., sensors with recognition algorithms are likely to exhibit some form of learning behavior, with error rates decreasing with the number of observations. Further, we assume that the operator's search budget is large; otherwise, a dynamic programming approach might be suitable to this particular problem. It will be shown that for relatively small search budgets, the adaptive algorithm we propose is not the best choice.

C. SINGLE STATIC TARGET (*SST*) MODEL

1. Problem Statement

Again, consider an AOI and a particular TOI for which the operator is searching. This time, however, we are concerned with a TOI that is stationary, or *static*, somewhere within the AOI. Operational TOIs for this scenario might include, for example, enemy insurgents in hiding or a downed friendly aircraft. A *sequential eliminating procedure* is well suited for this particular application. In practice, any situation where the TOI remains static in a timescale that is larger than that of the search process fits the framework of this model. A sequential eliminating procedure attempts to isolate, from among several candidate *systems*, one particular desired system—the "objective." During a particular stage of a sequential eliminating procedure, all candidate systems are examined and ranked in order of their likelihood of being the objective. Each system is then compared to the *most likely objective*—the system ranked highest for that stage—by means of the ratio of their likelihood ratios (i.e., their *odds ratio*). Any system whose odds ratio fails to meet a certain threshold (which we define in Chapter III) is permanently removed (*eliminated*) from the set of candidates. If all systems meet the threshold during a particular stage, then all those systems remain in the set of candidates (this is referred to as the *continuation region*). The procedure advances to the next stage, using the updated candidate set. The process continues until only one system remains in the set of candidates, and that system is declared the winner (in our case, the *systems* are the area-cells, and the *winner* is the area-cell containing the TOI). The operational dilemma for this scenario is to make, as quickly as possible, a proper determination of TOI location, again based solely on the noisy sensor observations.

2. Objective

For the *SST* model, we set forth to develop efficient criteria for the sequential eliminating procedure, which, when followed, result in determination of TOI location meeting certain operator-defined error tolerances.

3. Scope and Limitations

We once again restrict our study to the case of a single target and a single sensor.[1] For the *SST* model, however, the TOI is assumed to be stationary, at least for the duration of the search period. The same assumptions made in the *SMT* model regarding sensor

[1] As an exception, we treat a brief introductory case (case 1) in which each area-cell either contains or does not contain a TOI, independent of all other area-cells.

sensitivity and specificity are relevant to the *SST* model, namely that the values are treated as fixed for each area, and that there is no correlation among sensor observations. Other assumptions made without loss of generality will be noted in Chapter III.

THIS PAGE INTENTIONALLY LEFT BLANK

II. LITERATURE REVIEW

This chapter discusses previous literature and research relevant to this thesis, and consists of four sections. We begin with an overview of some types of sensors currently in use or in development for unmanned aerial systems (UAS), as well as a brief survey of some research and literature related to search theory. An introduction to some ideas in the field of stochastic search and optimization follows, with special attention paid to the concept of stochastic approximation. We continue with a discussion on sequential analysis as a primer for the second model of this thesis. Finally, we present some conclusions and a justification for the research of this thesis.

A. BACKGROUND

1. Sensors and Complex Environments

The nature of recent conflicts has imparted a two-fold effect on the employment of UASs. First, it has caused the primary areas of operation to migrate into areas in which it is difficult to operate. Second, it has placed increased importance on the technological development of airborne Intelligence, Surveillance, and Reconnaissance (ISR) sensors to counter both asymmetric and conventional threats. Primary missions areas for tactical UASs in Iraq and Afghanistan include point surveillance, target following, area search, route reconnaissance, and Improvised Explosive Device (IED) detection (Owen, Martin, & Carriger, 2005). Missions flown by Pioneer, Scan Eagle, and Shadow UASs normally service a list of targets provided by intelligence units. These target lists are typically comprised of, for example, suspected insurgent safe houses, suspected weapons caches and mortar points of origin as well as direct support for raids, patrols, convoys and other operations (Reber, 2007). Additionally, recent research efforts have explored the use of UASs in the detection of possible chemical or biological plumes (Scheidt, 2008). Modern conventional and emergent asymmetric threats have indeed shaped a challenging battle-space to frame these missions.

Geographic areas that operators consider inhospitable or undesirable for the employment of UASs include terrain prevalent in current regions of major conflict. A geometrically diverse urban canopy and a cluttered, mountainous border crossing are two examples of areas that many consider exceptionally challenging for the operation of UASs. We refer to these locations collectively as *complex environments*. The challenges faced by operators and customers of UASs associated with these complex environments include, for example, variable levels of autonomy, collision avoidance, wind gusts and turbulence, unreliable wireless communications, stealth, power and energy management, and portability (Dodd & Apopei, 2007).

An elementary characterization of UAS-borne sensors in operation and under development today is via their *sensitivity* and *specificity,* two terms adopted from the binary classification test as measures of performance for discriminatory sensors (Kress, Szechtman, and Jones, 2008). The sensitivity of a sensor is a measure of its ability to correctly detect a real target, whereas a sensor's specificity is its ability to correctly reject (i.e., not detect) everything that is not a target of interest. Both of these characteristics are measured as probabilities, and lead to complementary sensor error rates. False negative (*miss*) rates are expressed algebraically as 1–sensitivity, and false positive (*false alarm*) rates are expressed as 1–specificity. Examples of sensors that may be characterized by both sensitivity and specificity include chemical and biological plume detectors, Inverse Synthetic Aperture Radar (ISAR), Synthetic Aperture Radar (SAR), multispectral and hyperspectral imaging sensors, and other Coherent Change Detection (CCD) and recognition-based sensors (Suter, 2005).

Determining a sensor's sensitivity and specificity for any region is not a trivial matter. Environmental factors such as wind, temperature, humidity, ambient light, and atmospherics, as well as physical makeup of the target compared to its surroundings, target and searcher motion, and line-of-sight considerations are all capable of affecting sensor performance (Calhoun, et al., 2007). For certain Human-in-the-Loop (HITL) systems, the added complicating factor of operator recognition differential (RD) is often subjective and very difficult to quantify. Additionally, certain sensors are susceptible to performance degradation over time, possibly decreasing both sensitivity and specificity.

2. Search Theory and Mission Planning

Since operators can afford neither infinite dwell time nor infinite sensors, a model must be developed that allocates assets, constrained by a particular search budget, in a manner that optimizes certain measures of performance and measures of effectiveness (MOP/MOE). Much of the research done to date on sensor allocation has its roots in the ideas put forth by Koopman in his pioneering report, *Search and Screening*, penned in 1946 and declassified in 1958.

Benkoski, Monticino, and Weisinger (1991) give a survey of literature published on search theory up until 1991. Their discussion covers problems with non-cooperative targets, as opposed to cooperative or rendezvous problems. The class of non-cooperative target problems includes both those having to do with passive targets (one-sided search) and those concerned with evasive targets (search games). They break down search problems by time and space (discrete versus continuous), target motion (stationary versus moving), and constraints on searcher motion (paths versus search effort allocation). The authors also discuss additional extensions, including multiple searchers and targets, uncertain detection probabilities, and varying objective functions.

One of the most widely cited treatments, and one whose motivation relates most closely to the focus of this thesis, is that proposed by Washburn (1983). His study concerned the application of an iterative Forward and Backward (FAB) algorithm, originally put forth by Brown (1977), to compute optimal (in the sense of maximizing probability of target detection) search plans when the motion of the target is modeled by a discrete space and time Markov chain with known transition matrix. The FAB algorithm is also the tool of choice for Dambreville and Le Cadre (1999) to allocate search effort in the case where search assets renew with generalized linear constraints. Oshumi (1991) tackles a similar problem to that of Washburn, but in continuous time where target motion is described by stochastic differential equations, rather than by a Markov process.

Prior to Washburn's work, moving target problems could only be solved in certain cases. One particular case is that in which target motion is conditionally deterministic with a factorable Jacobian. In this case, one can reduce the to a stationary-target problem and solve it via stationary-target techniques (Stone, 1977; Pursiheimo, 1976; and Iida, 1972). The other case is one in which the number of area-cells is small, despite *a priori* knowledge of the associated optimality conditions (Lions, 1971; Hellman, 1972; Saretsalo, 1973; Pollock, 1970; & Dobbie, 1974).

Taking the idea of a non-cooperative target a step further, some authors have considered targets that take evasive action. Such problems rarely lend themselves to analytic solutions; in such cases, simulation may provide a suitable alternative (Washburn, 1989). Another approach utilizes the *minimax* strategy of game theory to maximize the searcher's probability of detecting an evasive target (Dambreville and Le Cadre, 2001). One possible drawback to the minimax strategy is the potentially prohibitive computational cost involved upon introduction of varied strategies for multiple searchers or targets. Carl (2003) chose the former approach in his thesis studying the search for German U-Boats in the Bay of Biscay during World War II, using agent-based simulation to evaluate Allied search plans.

DelBalzo and Hemsteter (2002) present a genetic algorithm approach to the evasive target problem. They show that, in general, an evasively maneuvering target as compared to a randomly patrolling target reduces the cumulative detection probability (CDP) in sonar search dramatically, since counter-detection ranges are typically greater than detection ranges. Their analysis covers several combinations of platforms and sensors in a simulated environment. Their Genetic Range-dependent Algorithm for Search Planning (GRASP) and associated joint tactics exploit evasive target maneuvers and provide increased CDP over non-joint tactics.

Whereas the typical objective of a *detection* search is to maximize probability of detection, the objective of a *surveillance* search is to maximize the probability of detecting the target at a specific time or in a specific region. Similarly, the object in *whereabouts* search is to localize a target to within one of a finite number of cells (Benkoski, Monticino, & Weisinger, 1991). In this case, searcher success is achieved by either detecting the target, or, if the target is not detected, by correctly guessing the cell containing the target. Tognetti (1968) and Kadane (1971) treat the scenario of whereabouts search against a stationary target. By showing that solving a whereabouts search is equivalent to solving a finite number of optimal detection search problems, Stone and Kadane (1981) make general the earlier results to encompass the moving target problem. Finally, Tierney and Kadane (1983) provide necessary optimality conditions and an algorithm that constructs search plans for the surveillance search problem against a Markovian target with known transition matrix.

Dell, Eagle, Santos, and Martins (1996) formulate a discrete time, path optimization problem for multiple searchers. They utilize a branch-and-bound algorithm and six heuristics for solving such problems. In a related approach, Sato and Royset (2008) develop a specialized branch-and-bound algorithm and a Lagrangian relaxation-based bounding technique to solve problems where the searcher is constrained by consumption limits of several resources. In their problem, the searcher knows both the initial target distribution and its Markovian transition matrix.

The problem addressed by Zhang and Chen (2006) deals with multiple imperfect sensor allocation against multiple targets in discrete time, where the overall goal is to minimize target location error. In their model, estimated target position is represented through a probability grid updated dynamically by belief states based on sensor input. Sensor errors are inherent in the Gaussian signal strength inputs of the sensors; however, they do not deal directly with sensitivity and specificity in calculating allocations that are optimal in the sense of minimizing incorrect determinations.

In his thesis, Lohr (1992) discusses *Area Motion Search*, a hybrid of classical search and detection theory models of *Exhaustive Search* and *Random Search*. In this model, target motion is random in continuous time, searcher motion is systematic in continuous time, and detection opportunities in non-overlapping time periods are probabilistically independent. Again, sensor error rates are not considered.

Peot et al. (2005) suggest a probabilistic roadmap approach to the urban UAS routing problem which maximizes the utility of sensing actions for a given collection strategy, where the benefit is modeled as the cumulative probability of detection, recognition, or identification during the period that the target is observed. A penalty cost due to exposure to threats or navigational hazards in the environment offsets the benefit. While the robust tool is one that maximizes sensing actions with respect to sensor dwell, communications, Line of Sight (LOS), and UAS flight kinematic constraints, the authors do not address sensitivity and specificity of the sensors.

Yan and Blankenship (1987) propose a list of tasks for a detection search, and emphasize the non-trivial nature of each step. The steps they outline are highly applicable to our study:

1. Compute a prior distribution of target location.
2. Obtain a good estimate of sensor capabilities.
3. Determine a detection (misdetection) function.
4. Develop a search plan and estimate its success probability.
5. Update the posterior target distribution from search feedback.
6. Evaluate search effectiveness (Yan and Blankenship, 1987).

Step 5 is often referred to as the *Search Control Problem,* and, for Yan and Blankenship, involves determining a search path that minimizes the target survivability up to a certain time. They solve the problem on a simplified search model, in discrete time and space, by embedding the *Dual Estimation Problem* (Yan & Blankenship, June 1987) into their *Ordered Search Algorithm,* a best-first search algorithm. The resultant *Optimal Detection Search Algorithm* (ODSA) describes the real model more precisely. ODSA not only updates target distribution at the beginning of each time step to hone the accuracy of the *Search Estimation Problem* (see Step 4 above); it also finds an optimal path of the *Search Control Problem*. By applying an efficient heuristic to the *Ordered Search Algorithm,* Yan and Blankenship show convergence to optimal paths, while expanding only about one ninth of the nodes expanded by an exhaustive search, thus fulfilling the evaluation of search effectiveness (see Step 6 above).

As noted in Step 1, the ability to construct a prior distribution of target position is of interest. One possible source for determining the initial target probability mass function is a tool known as *Threat Mapper*. Riese (2006) developed the software tool, which leverages the robustness and availability of geospatial information systems (GIS) and fuses historical data to aid analysts and forces in making spatial forecasts to support intelligence operations. The planner enters locations of past events of interest (see Figure 1, wherein historical events are represented by red triangles), and areas of spatial similarity are determined and used as a forecast for future events. The output is a color-coded map of absolute spatial similarity, as determined by user input characteristics (see Figure 2, where red areas are of high likelihood, and blue areas are of low likelihood), and can be normalized to create what can be used as a probability map.

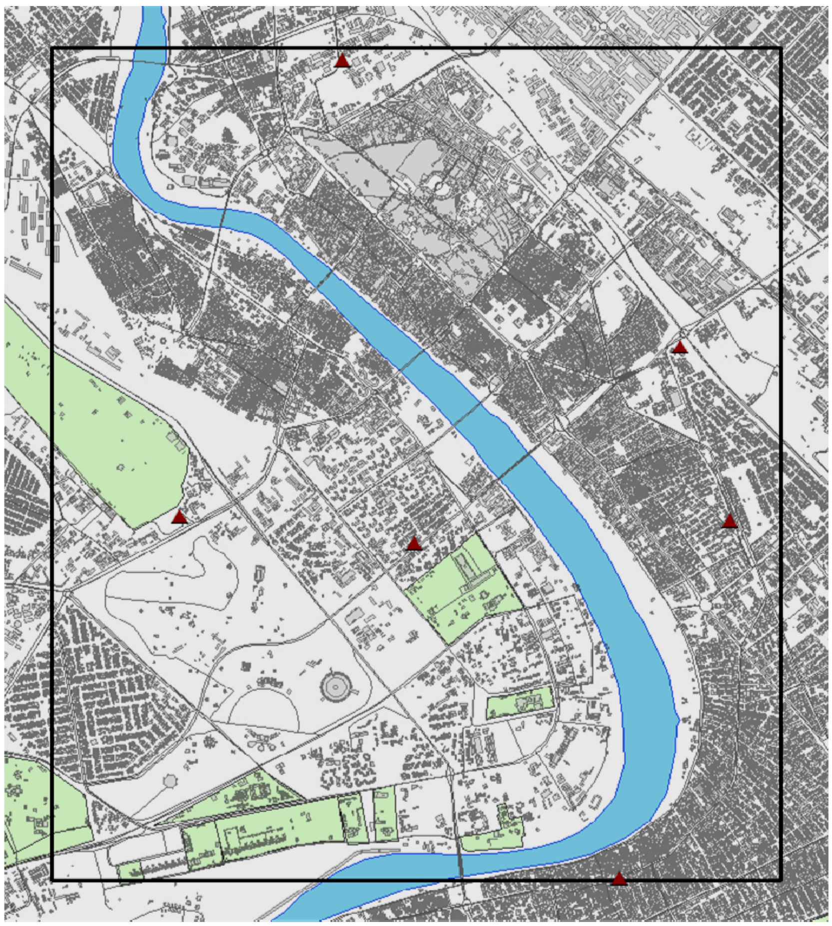

Figure 1. Example *Threat Mapper* input (From Riese, 2008).

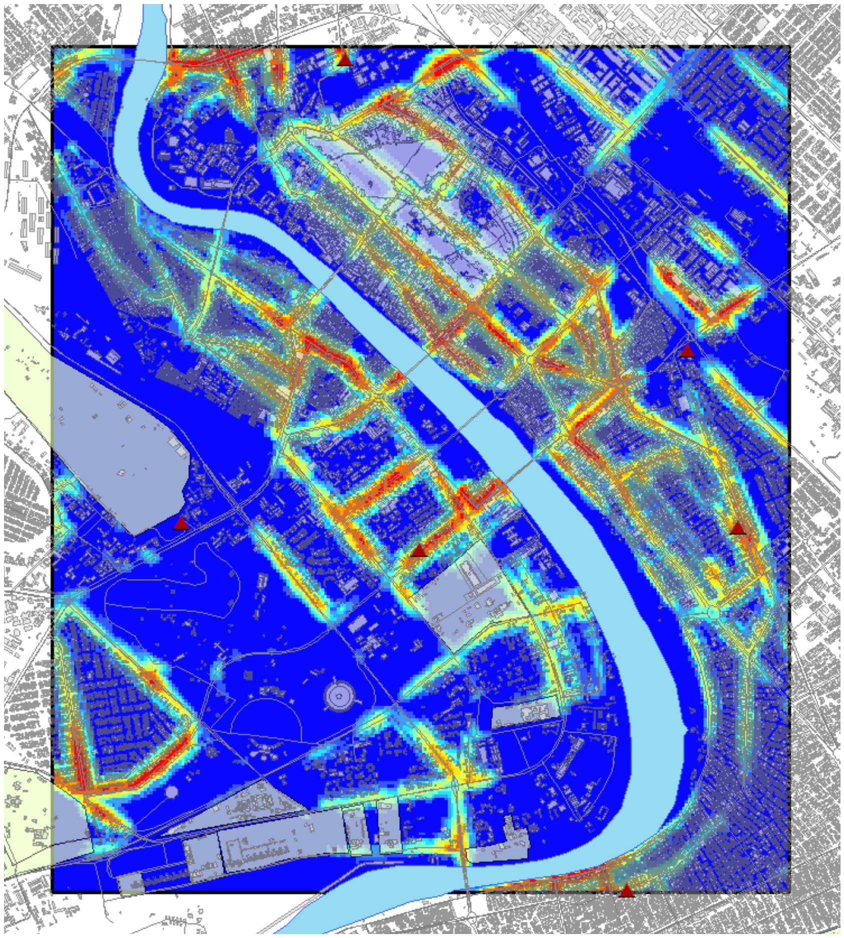

Figure 2. Example *Threat Mapper* output (From Riese, 2008).

B. STOCHASTIC SEARCH AND OPTIMIZATION

Whereas much of the previous research associated with the UAS routing problem has focused on classical search theory as explained in Section C of this chapter, the noise and uncertainty associated with the complex environments described above necessitate a robust tool. Given the complexity of many real-world problems faced by industry and government today, along with the inherent uncertainty in the information that might be available to the problem solver, *stochastic search and optimization* models have been playing a growing role (Spall, 2003).[2] In stochastic optimization, there are generally two possible sources of uncertainty. In the first case, random noise is present in the measurement of either a loss function, which is a scalar measurement summarizing the performance of the system for a given value of the vector of the "adjustables," or its gradient function. In the second case, there is a random choice made in the search direction as the algorithm iterates toward a solution. For a given problem, there is also the possibility that noise is present due to both of the aforementioned cases. In the problem presented in this thesis, the sources of noise are the misclassification errors inherent to the sensors, and the uncertainty associated with the movement of a target.

One of the cornerstones of stochastic search and optimization is the idea of stochastic approximation (SA) (see Kushner & Yin, 2003). SA algorithms are iterative methods of finding extremes or roots of functions whose values cannot be calculated directly, but instead must be approximated based on noisy observed values. For example, let us start by considering a real function *g*, and suppose the goal is to find the value θ^* such that $g\left(\theta^*\right)=0$. Assume, for simplicity, that $g\left(\theta\right)>0$ for $\theta<\theta^*$ and that $g\left(\theta\right)<0$ for $\theta>\theta^*$. The recursive procedure is

$$\theta_{n+1}=\theta_n+\varepsilon g(\theta_n)$$

for $\varepsilon>0$. If $\theta_n<\theta^*$ then $g\left(\theta_n\right)>0$, meaning that $\theta_{n+1}>\theta_n$, and hence θ_{n+1} moves in towards θ^*. If $\theta_n>\theta^*$ then $g(\theta_n)<0$, so that $\theta_{n+1}<\theta_n$, and θ_{n+1} moves to the left, approaching θ^*.

Robbins and Monro (1951) extended the above procedure to the case where the function *g* is unknown, but can be estimated via noisy observations. The recursion is

$$\theta_{n+1}=\theta_n+\varepsilon_n Y_n$$

where $\varepsilon_n>0$, $\varepsilon_n\to 0$, $\sum_n \varepsilon_n=\infty$, and Y_n are noisy observations of $g\left(\theta_n\right)$. More precisely, $Y_n=g\left(\theta_n\right)+\delta$, with $E\left[Y_n \mid Y_i,\theta_i,i<n,\theta_n\right]=g\left(\theta_n\right)$, and where the error term δ

[2] Here, the term "search" is referring to the algorithmic approach to finding an optimal solution, as opposed to attempting to locate a target, which has been the definition referred to thus far.

has finite variance. (The conditioning elements, $Y_i, \theta_i, i < n, \theta_n$, comprise the history up to stage *n*.) In this case, the recursion can be written as

$$\theta_{n+1} = \theta_n + \varepsilon_n g(\theta_n) + \varepsilon_n \cdot \delta,$$

suggesting that the effect of the error term δ vanishes as $n \to \infty$ due to the finite error variance and the conditions imposed on ε_n.

Soon after Robbins and Monro (1951) introduced their algorithm, Kiefer and Wolfowitz (1952) built upon it by injecting a second sequence of positive step sizes, which are used to estimate the derivative of the function of true values via the difference between the observed values and the new step sizes. They showed that if both sequences of step sizes fulfill certain bounds, and the functions of the noisy and true values satisfy certain conditions, then the observed values converge in probability to the true value. It is a concept similar to that of the Robbins-Monro and Kiefer-Wolfowitz algorithms that is at the heart of the stochastic approximation algorithm used in the *SMT* model of this thesis.

C. SEQUENTIAL ANALYSIS

In the second situation of this thesis, the operator is concerned with quickly determining the presence or absence of a target subject to certain type-I and type-II error tolerances, which are specified by the operator. The goal is to stop the search in the least expected amount of time, subject to the error bounds. This problem is intrinsically sequential, as it deals with the fixed precision estimation of a parameter in the presence of an unknown nuisance parameter. The theory behind *sequential analysis* is therefore a well-suited solution approach in this case.

Siegmund (1985) is the classic reference in this field, and deals primarily with sequential hypothesis testing and related problems of estimation. In many of these cases, a fixed sample solution exists and one employs sequential methods in order to achieve some greater efficiency in the solution. For example, consider the case where one wishes to infer, on the basis of a random sample, whether the proportion of defective items in a large batch exceeds some value p_0. Assume that the inference will be based on the number S_m of defectives in a random sample of size *m*. If *m* is a small proportion of the batch size, then S_m has (approximately) a binomial distribution with mean *mp*, where *p* is the true proportion of defectives in the batch, and a reasonable rule to test the hypothesis

$$\mathcal{H}_0 : p \leq p_0 \text{ against } \mathcal{H}_1 : p > p_0$$

is to

$$\text{reject } \mathcal{H}_0 \text{ if } S_m > r$$

for some constant *r*. If the sample is drawn sequentially, and for some value *k* less than *m*, the value of S_k already equals *r*, one could stop sampling immediately and reject $\mathcal{H}_0$. More formally, let *T* denote the smallest value of *k* for which $S_k = r$ and put $T' = \min(T, m)$. Consider the procedure that stops sampling at the random time T' and decides that $p > p_0$ if and only if $T \leq m$. If one considers these procedures as tests of $\mathcal{H}_0$ against $\mathcal{H}_1$, their rejection regions, namely $\{T \leq m\}$ and $\{S_m \geq r\}$, are the same events, and hence the two tests have the same power function. Since the test which stops at random time T' never takes more observations and may take fewer observations than the fixed sample test, it has a reasonable claim to be regarded as more efficient (Siegmund, 1985, p. 2). Siegmund acknowledges additionally that sequential methods are a natural choice for parameter estimation problems, such as the *SST* model of this thesis.

Malone (2004) treats ranking and selection procedures for both Bernoulli and multinomial systems. These Bernoulli ranking and selection procedures are related to our problem, since the sensors sample from a Bernoulli distribution for each *system* (area-cell), with parameter that depends on whether the target is present or absent in that area-cell. In her thesis, however, each system has *unknown* Bernoulli parameter $a_1, ..., a_m$, and the goal is to select the system with the largest a. She applies fully sequential procedures to Bernoulli data for terminating solutions, and significant savings in total observations are realized for two to five systems, when one desires to detect small differences between competing systems.

Wieland and Nelson (2004) present a sequential, eliminating procedure for selecting the best system in a *single-factor* Bernoulli-response experiment with an odds-ratio indifference zone. Similar to Malone, in their case, “best” refers to the system with largest probability of success on a given trial. Recall, in subtle contrast, the problem considered in this thesis. Consider an AOI comprised of m area-cells, labeled $AC_1,...,AC_m$. Assume, without loss of generality, that a target is present within AC_1. Let a_i be the sensor’s *sensitivity* for cell i, and let $1-b_i$ be the *specificity* of the sensor for cell i (see Chapter III for definitions of these terms). In AC_1, we sample from a Bernoulli distribution with parameter a_1. In AC_i, $i=2,...,m$, we sample from a Bernoulli distribution with parameter b_i. Given that a_i and b_i are known for $i=1,...,m$, the problem is that of determining the area-cell from which samples are drawn from a Bernoulli distribution with parameter a.

D. LITERATURE REVIEW CONCLUSIONS

Upon review, the nature of recent sensor employment in complex environments, with consideration to the characteristics of emergent sensor technology, necessitates models that provide efficient allocation of scarce sensors in order to provide sensor operators and operational planners with the availability and flexibility required on today’s battlefields. We acknowledge that significant research has been conducted on search effort allocation against a moving target. Nevertheless, we hope to offer genuinely new insight by framing the problem in this operational context, in discrete time and space, while considering both sensor sensitivity and specificity, and through use of an adaptive algorithm based upon stochastic approximation to determine steady-state location distribution of a Markovian target with unknown transition matrix. Additionally, the research to date in the field of sequential analysis has only considered eliminating procedures that attempt to find the “best” Bernoulli system—typically the one with maximum (unknown) parameter value. In contrast, we wish to find the system whose parameter is most likely to be equal to a known value, which is different for each system and may not be the maximum. This is an operationally relevant problem for static targets, which we set forth to solve efficiently with a sequential eliminating procedure that is guaranteed to meet type-I and type-II error thresholds.

THIS PAGE INTENTIONALLY LEFT BLANK

III. MODEL DEVELOPMENT

In this chapter, we discuss the development and formulation of the models used in this thesis. We start by introducing some theory and details of the *SMT* model, then proceed to outline the stochastic approximation algorithm used to provide solutions. We then follow a similar pattern for the *SST* model and its sequential eliminating procedure.

A. *SMT* MODEL

1. Basic Framework

Suppose that the area of operational interest (AOI) is partitioned into *m* area-cells. Let $\pi = (\pi_1, ..., \pi_m)$ be the steady-state distribution of the target. The main goal is to estimate π by employing an imperfect sensor to look into the area-cells. The sensor is characterized by its sensitivity and specificity. For each area-cell,

$$a_i = P(\text{sensor indicates detection in area cell } i \mid \text{target is in area-cell } i)$$

is the sensitivity, and $1-b_i$ is the specificity, where

$$b_i = P(\text{sensor indicates detection in area cell } i \mid \text{target is not in area-cell } i).$$

We assume that the sensitivity and specificity are known. Suppose $a_i > b_i$ (otherwise we can reverse the sensor cue, meaning that a "target present" indication is interpreted as "target absent," and vice versa).

Consider area-cell i. Let $X_{i,1}, X_{i,2}, ...$ be independent and identically distributed (IID) random variables that describe the sensor observations, where $X_{i,j}=1$ if the sensor returns a *detection (hot)* signal in the j'th look into area-cell i, and $X_{i,j}=0$ if the sensor returns a *no detection (cold)* signal. Thus $\{X_{i,j}\}_{j=1}^{\infty}$ is a collection of Bernoulli IID random variables, with $P(X_{i,1}=1) = (1-\pi_i)b_i + \pi_i a_i$. We defer to further study the option to relax the IID assumption in order to allow for some correlation among sensor observations.

The decision variables are $p_1,\ldots,p_m$, the fraction of the search budget allocated to each area-cell, where $\sum_i p_i = 1$ and $p_i \geq 0$. The search budget is described by *n*. In this work, we are interested in determining efficient search allocations for *n* large. It is therefore reasonable to assume that the TOI is already in steady-state.

2. Large Search Budget Results

Let $\bar{X}_i(p_i n)$ be the fraction of *detections* in area-cell i by the time of the $\lfloor p_i n \rfloor$'th look (in what follows we work with $p_i n$ instead of $\lfloor p_i n \rfloor$; since our results hold for *n* large, they continue to be true when the integrality condition is enforced, by working with a sequence that goes to infinity). By the Strong Law of Large Numbers, we know that

$$\frac{\bar{X}_i(p_i n) - b_i}{a_i - b_i} \to \pi_i$$

with probability 1 as $n \to \infty$ (Ross, 1996, p. 41).

We choose to minimize the largest absolute error. Thus, by standard results in large deviations theory (Dembo and Zeitouni, 1998) we have

$$\begin{aligned} & P\left(\left|\frac{\bar{X}_i(p_i n) - b_i}{a_i - b_i} - \pi_i\right| > \varepsilon\right) \\ &= P\left(\left|\bar{X}_i - \left(a_i \pi_i + b_i\left(1 - \pi_i\right)\right)\right| > \left(a_i - b_i\right)\varepsilon\right) \\ &\approx \exp\left(-p_i n \min\left\{I_i\left(b_i + \left(a_i - b_i\right)\left(\pi_i + \varepsilon\right)\right), I_i\left(b_i + \left(a_i - b_i\right)\left(\pi_i - \varepsilon\right)\right)\right\}\right), \end{aligned} \tag{1.1}$$

where $I_i(\cdot)$ is the large deviations rate function (see Dembo and Zeitouni, 1998, p. 4). Also,

$$\begin{aligned} \max_i P\left(\left|\frac{\bar{X}_i(p_i n) - b_i}{a_i - b_i} - \pi_i\right| > \varepsilon\right) &\leq P\left(\max_i \left|\frac{\bar{X}_i(p_i n) - b_i}{a_i - b_i} - \pi_i\right| > \varepsilon\right) \\ &\leq m \max_i P\left(\left|\frac{\bar{X}_i(p_i n) - b_i}{a_i - b_i} - \pi_i\right| > \varepsilon\right). \end{aligned} \tag{1.2}$$

Combining the outcomes of (1.1) and (1.2) results in

$$\frac{1}{n}\log P\left(\max_i\left|\frac{\overline{X}_i(p_i n)-b_i}{a_i-b_i}-\pi_i\right|>\varepsilon\right) \qquad (1.3)$$
$$\to -\min_i p_i \min\left\{I_i\left(b_i+\left(a_i-b_i\right)\left(\pi_i+\varepsilon\right)\right), I_i\left(b_i+\left(a_i-b_{ii}\right)\left(\pi_i-\varepsilon\right)\right)\right\},$$

as $n \to \infty$.

For a non-degenerate Bernoulli random variable $X_{i,1}$ with mean μ_i, the large deviations rate function I_i is given by

$$I_i\left(\gamma_i\right)=\gamma_i\log\left(\frac{\gamma_i}{\mu_i}\right)+\left(1-\gamma_i\right)\log\left(\frac{1-\gamma_i}{1-\mu_i}\right),$$

for $0<\gamma_i<1$.

The optimal allocation solves

$$\max_{p_1,\ldots,p_m}\min_i\ p_i\min\left\{I_i\left(b_i+\left(a_i-b_i\right)\left(\pi_i+\varepsilon\right)\right), I_i\left(b_i+\left(a_i-b_{ii}\right)\left(\pi_i-\varepsilon\right)\right)\right\}$$

s.t.

$$\sum_{i=1}^{m} p_i \le 1$$
$$p_i \ge 0$$

and makes equal the exponential decay rates that appear in the right hand side of Equation (1.1), for each area-cell i. The (unique) solution is given by

$$p_i^* = \frac{\min\left\{I_i\left(b_i+\left(a_i-b_i\right)\left(\pi_i+\varepsilon\right)\right), I_i\left(b_i+\left(a_i-b_i\right)\left(\pi_i-\varepsilon\right)\right)\right\}^{-1}}{\sum_{j=1}^{m}\min\left\{I_j\left(b_j+\left(a_j-b_j\right)\left(\pi_j+\varepsilon\right)\right), I_j\left(b_j+\left(a_j-b_j\right)\left(\pi_j-\varepsilon\right)\right)\right\}^{-1}},$$

where we choose ε sufficiently small so that the argument inside each logarithm is non-negative. The steady-state distribution π is unknown, so it is replaced by the standard, sampling-based estimator that is obtained as the stochastic approximation algorithm (described in the next section) makes progress.

3. Stochastic Approximation Algorithm

We first present the algorithm, with the intuition behind it following immediately thereafter.

Initialization. Let $0 < \tilde{X}_{i,0} < 1$ be our initial guess of $b_i + (a_i - b_i)\pi_i$, and set $\gamma^+_{i,0} = \tilde{X}_{i,0} + (a_i - b_i)\varepsilon$, and $\gamma^-_{i,0} = \tilde{X}_{i,0} - (a_i - b_i)\varepsilon$. The initial guesses of the rate functions are

$$I^+_{i,0} = \gamma^+_{i,0}\log\left(\frac{\gamma^+_{i,0}}{\tilde{X}_{i,0}}\right) + \left(1-\gamma^+_{i,0}\right)\log\left(\frac{\left(1-\gamma^+_{i,0}\right)}{\left(1-\tilde{X}_{i,0}\right)}\right),$$

and

$$I^-_{i,0} = \gamma^-_{i,0}\log\left(\frac{\gamma^-_{i,0}}{\tilde{X}_{i,0}}\right) + \left(1-\gamma^-_{i,0}\right)\log\left(\frac{\left(1-\gamma^-_{i,0}\right)}{\left(1-\tilde{X}_{i,0}\right)}\right).$$

The initial guess of the optimal allocation is

$$p_{i,0} = \frac{\min\left\{I^+_{i,0}, I^-_{i,0}\right\}^{-1}}{\sum_{j=1}^{m}\min\left\{I^+_{j,0}, I^-_{j,0}\right\}^{-1}}.$$

Finally, set $\ell = 0$.

Algorithm *SA*.

1. Generate a replicate ξ from the probability mass function $p_{1,\ell}, \ldots, p_{m,\ell}$.

2. Update sample sizes: $\lambda_{\xi,\ell+1} = \lambda_{\xi,\ell} + 1$, and $\lambda_{i,\ell+1} = \lambda_{i,\ell}$ for $i \neq \xi$.

3. Generate a sample from area-cell ξ, (say) X_{ξ,λ_ξ}, from a Bernoulli with parameter $a_\xi \pi_\xi + b_\xi\left(1-\pi_\xi\right)$.

4. Update $\tilde{X}_{\xi,\ell+1}, \gamma^+_{\xi,\ell+1}, \gamma^-_{\xi,\ell+1}, I^+_{\xi,\ell+1}$, and $I^-_{\xi,\ell+1}$:

$$\tilde{X}_{\xi,\ell+1} = \tilde{X}_{\xi,\ell} + \frac{1}{\lambda_{\xi,\ell+1}}\left(\tilde{X}_{\xi,\lambda_\xi} - \tilde{X}_{\xi,\ell}\right).$$

Set $\gamma^+_{\xi,\ell+1} = \tilde{X}_\xi + \left(a_\xi - b_\xi\right)\varepsilon$, and $\gamma^-_{\xi,\ell+1} = \tilde{X}_\xi - \left(a_\xi - b_\xi\right)\varepsilon$. Let

$$I^{+}_{\xi,\ell+1} = \gamma^{+}_{\xi,\ell+1} \log\left(\frac{\gamma^{+}_{\xi,\ell+1}}{\tilde{X}_{\xi,\ell+1}}\right) + \left(1-\gamma^{+}_{\xi,\ell+1}\right)\log\left(\frac{\left(1-\gamma^{+}_{\xi,\ell+1}\right)}{\left(1-\tilde{X}_{\xi,\ell+1}\right)}\right),$$

and

$$I^{-}_{\xi,\ell+1} = \gamma^{-}_{\xi,\ell+1} \log\left(\frac{\gamma^{-}_{\xi,\ell+1}}{\tilde{X}_{\xi,\ell+1}}\right) + \left(1-\gamma^{-}_{\xi,\ell+1}\right)\log\left(\frac{\left(1-\gamma^{-}_{\xi,\ell+1}\right)}{\left(1-\tilde{X}_{\xi,\ell+1}\right)}\right).$$

For $i \neq \xi,$ set $\tilde{X}_{i,\ell+1},$ $\gamma^{+}_{i,\ell+1} = \gamma^{+}_{i,\ell},$ $\gamma^{-}_{i,\ell+1} = \gamma^{-}_{i,\ell},$ $I^{+}_{i,\ell+1} = I^{+}_{i,\ell},$ and $I^{-}_{i,\ell+1} = I^{-}_{i,\ell}.$

5. Update $p_{\cdot,\ell+1}$:

$$p_{i,\ell+1} = \frac{\min\left\{I^{+}_{i,\ell+1}, I^{-}_{i,\ell+1}\right\}^{-1}}{\sum_{j=1}^{m} \min\left\{I^{+}_{j,\ell+1}, I^{-}_{j,\ell+1}\right\}^{-1}}.$$

6. Increase $\ell \leftarrow \ell + 1$ and go back to 1.

To see why our algorithm leads to the optimal allocations, let $\theta_{i,\ell} = \lambda_{i/\ell} / \ell$ be the fractional allocations in stage ℓ of the algorithm. Hence, step 2 of the algorithm can be expressed as $\theta_{i,\ell+1} = \theta_{i,\ell} + \left(J\left(\xi_\ell = i\right) - \theta_{i,\ell}\right) / \left(\ell+1\right),$ where ξ_ℓ is the ℓth replicate of ξ generated in step 1 of the algorithm, and $J\left(\cdot\right)$ is the indicator function. The recursion for $\theta_{i,\ell+1}$ can be re-written as

$$\theta_{i,\ell+1} = \theta_{i,\ell} + \frac{1}{\ell+1}\left(p_i^* - \theta_{i,\ell}\right) + \varepsilon_\ell,$$

where

$$\varepsilon_\ell = \frac{1}{\ell+1}\left(J\left(\xi_\ell = i\right) - p_{i,\ell}\right) + \frac{1}{\ell+1}\left(p_{i,\ell} - p_i^*\right).$$

If the error ε_ℓ becomes small relative to the $\left(p_i^* - \theta_{i,\ell}\right)\big/\left(\ell+1\right)$ term, then $\theta_{i,\ell}$ follows, as $\ell \to \infty,$ the path of the solution of the ordinary differential equations

$$\theta_i' = p_i^* - \theta_i, \quad i = 1,\ldots,m,$$

which have p_i^* as the unique globally asymptotically stable point. This suggests that if the variability introduced by the error at each stage is sufficiently small

$\left(\text{i.e., } \operatorname{var}\left(\varepsilon_1\right)+\ldots+\operatorname{var}\left(\varepsilon_\ell\right)=o\left(\ell\right)\right)$, our algorithm provides fractional allocations that converge almost surely to the optimal allocations. A rigorous analysis of this approach is discussed in Kushner and Yin (2003, p.170).

B. *SST* MODEL

1. Case 1: Independent Area-cells with No Target or One Target

While not an operationally relevant or likely scenario, we treat the case where each area-cell either contains or does not contain a TOI, independent of all other area-cells, because it brings intuition about the single-target cases (case 2 and case 3) of this chapter.

<u>Procedure *SP*1</u>

Consider one area-cell, and suppose that the sensor operator prescribes an acceptable false-positive probability α and a false-negative probability β. The observations $X_1, X_2, \ldots$ are drawn from a Bernoulli random variable with parameter *a* if the area-cell contains a target, or from a Bernoulli random variable with parameter *b* if the area-cell does not contain a target. Let $S_n = X_1 + \ldots + X_n$ be the number of detections after n looks. For $0 \le p \le 1$, the likelihood function $p^{S_n}\left(1-p\right)^{1-S_n}$ can be used to determine whether the unknown Bernoulli parameter is a or b, because this likelihood is maximized by $p=a$ if the area-cell contains a target and by $p=b$ if it does not. Hence the likelihood ratio

$$\ell_n\left(x_1,\ldots,x_n\right)=\frac{a^{s_n}(1-a)^{n-s_n}}{b^{s_n}(1-b)^{n-s_n}}\to\infty$$

if a target is present, and $\ell_n\left(x_1,\ldots,x_n\right)\to 0$ if the area-cell is target-free, as $n\to\infty$. This suggests that a judicious policy is to stop sampling when the likelihood ratio crosses an upper threshold and declare the target present, or when the likelihood ratio crosses a lower threshold and declare the target absent. This approach may lead to an incorrect determination, but its probability can be prescribed *ab initio* by the end-user.

Define the hypotheses:

$\mathcal{H}_0$: Target absent from area-cell, and $\mathcal{H}_1$: Target present in area-cell.

Given our definitions of sensor sensitivity and specificity, these hypotheses are analogous to sampling from a Bernoulli distribution with parameter p where

$$\mathcal{H}_0 : p = b, \text{ or } \mathcal{H}_1 : p = a.$$

Define the stopping time

$$N = \inf\left\{n \geq 1 : \ell_n \notin (A, B)\right\}$$

for the threshold constants $A,\ B$ such that $-\infty < A < B < \infty$. Then we

$$\text{Reject } \mathcal{H}_0 \text{ if } \ell_N \geq B \text{ and Accept } \mathcal{H}_0 \text{ if } \ell_N \leq A.$$

For $0 \leq \alpha, \beta \leq 1$ prescribed by the end-user, the error probabilities are

$$\text{Type I error: } P\left(\ell_N \geq B \mid \mathcal{H}_0\right) = \alpha \text{ and Type II error: } P\left(\ell_N \geq A \mid \mathcal{H}_1\right) = \beta\,.$$

Taking logarithms, we can see that we reject $\mathcal{H}_0$ if

$$S_N \geq \frac{\log B}{\log\left(\frac{a}{b}\frac{1-b}{1-a}\right)} + N\frac{\log\frac{1-b}{1-a}}{\log\left(\frac{a}{b}\frac{1-b}{1-a}\right)}.$$

(We know the denominator is positive; recall our assumption that $a > b$, otherwise we can reverse the sensor cue) We accept $\mathcal{H}_0$ if

$$S_N \leq \frac{\log A}{\log\left(\frac{a}{b}\frac{1-b}{1-a}\right)} + N\frac{\log\frac{1-b}{1-a}}{\log\left(\frac{a}{b}\frac{1-b}{1-a}\right)}.$$

Siegmund (1985, p. 10) shows that

$$\alpha = P_0\left(\ell_N \geq B\right) \leq B^{-1}\left(1-\beta\right) \tag{1.4}$$

and

$$\beta = P_1\left(\ell_N \leq A\right) \leq A\left(1-\alpha\right). \tag{1.5}$$

Hence, given operator defined tolerances α and β, by setting

$$B = \frac{1-\beta}{\alpha}, \text{ and } A = \frac{\beta}{1-\alpha},$$

we are guaranteed to satisfy the error probability constraints. Moreover, it is shown in Siegmund (1985, p. 11) that if Equations (1.4) and (1.5) hold with equality, then this approach minimizes the expected number of looks until crossing either boundary. Although (1.4) and (1.5) generally do not hold with equality, the algorithm is guaranteed to meet the error criteria

Figure 3 displays two possible sample paths computed by $SP1$. In this illustration, the white area represents the state space for the total number of detections after n looks $\left(S_n\right)$, and the two parallel dashed lines represent the bounds. A path between the bounds is still undetermined; thus the procedure continues until the path exits via one of the bounds (hence, this area is known as the *continuation region*). A sample path that exits the upper bound results in a declaration of *target present* in the area-cell, whereas a path exiting via the lower bound produces a *target absent* declaration.

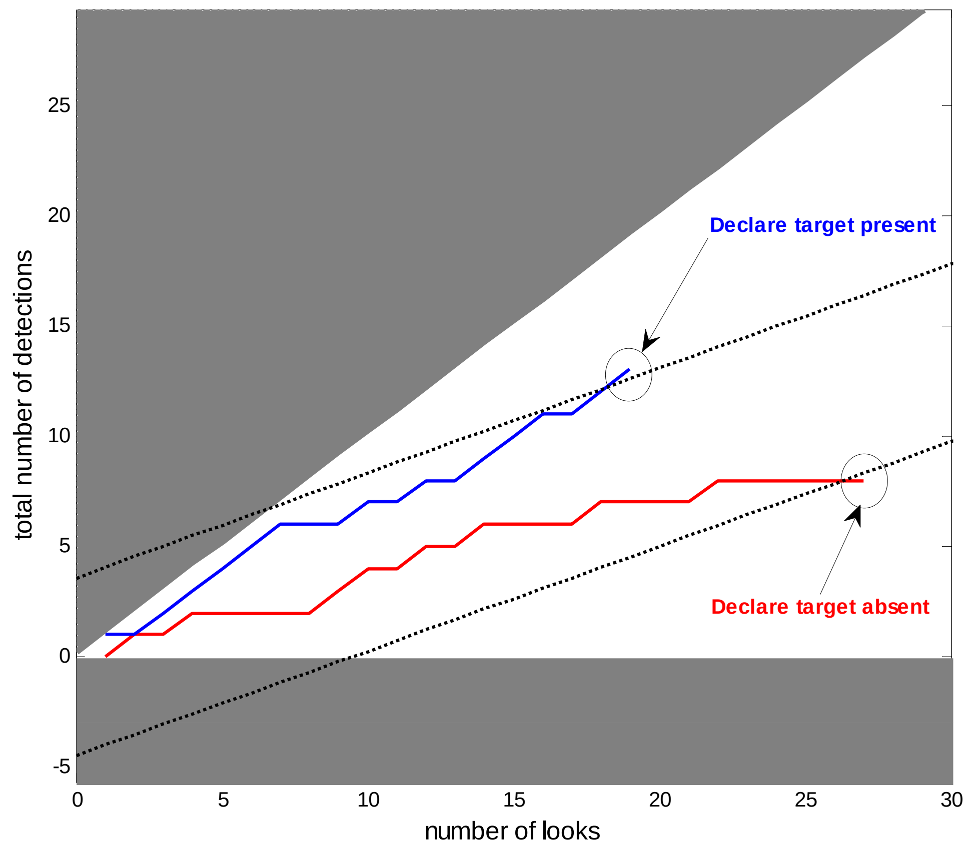

Figure 3. Example sample paths of the one-cell sequential procedure.

Next we discuss the case of a single target hidden among m area-cells, with $m > 1$. This case is difficult to analyze because knowledge about the presence/absence of a target in an area-cell yields light about the presence/absence of the target in other area-cells; i.e., the declarations about target absence/presence in each area-cell are no longer independent. In order to set the stage, we start with the case $m = 2$.

2. Case 2: Single Target and Two Area-cells

<u>Procedure *SP*2</u>

For this slightly more complicated case, consider (as a starting point) two area-cells, AC_0 and AC_1, with parameters (a_0, b_0) and (a_1, b_1), respectively, such that

$a_i > b_i$ for $i = 0,1$. For simplicity, we assume that both area-cells receive the same number of looks, and the goal is to terminate the inspection when there is enough evidence that the error bounds are met, i.e., we are confident to within our error tolerances of saying that the TOI is in a particular area-cell. The two possible errors are: (i) α is the probability that the target is determined to be in AC_1 when it is in AC_0, and (ii) β is the probability that the target is determined to be in AC_0 when in reality it is in AC_1. This leads to the hypotheses

$$\mathcal{H}_0 : \text{ Target located in } AC_0, \text{ and } \mathcal{H}_1 : \text{ Target located in } AC_1.$$

Let $S_{i,n}$ be the number of detections in AC_i, $i = 0,1,$ after n looks. Consider the ratio of likelihood ratios ℓ_n (often referred to as the "odds ratio"),

$$\ell_n = \frac{\ell_{1,n}}{\ell_{0,n}},$$

where

$$\ell_{i,n} = \frac{a_i^{s_{i,n}} \left(1-a_i\right)^{n-s_{i,n}}}{b_i^{s_{i,n}} \left(1-b_i\right)^{n-s_{i,n}}}.$$

We know that $\ell_{i,n} \to \infty$ if the target is present in AC_1 and $\ell_{i,n} \to 0$ if it is absent. Hence $\ell_n \to 0$ if the target is present in AC_0, and $\ell_n \to \infty$ otherwise. This suggests considering the stopping time

$$N = \inf\{n \geq 1 : \ell_n \notin \left(A, B\right)\}$$

for threshold constants A, B such that $-\infty < A < B < \infty$. Then we

$$\text{Reject } \mathcal{H}_0 \text{ if } \ell_N \geq B, \text{ and Accept } \mathcal{H}_0 \text{ if } \ell_N \leq A.$$

Taking logarithms, we can see that we reject $\mathcal{H}_0$ if

$$S_{1,N} \log\left(\frac{a_1}{b_1}\frac{1-b_1}{1-a_1}\right) - S_{0,N} \log\left(\frac{a_0}{b_0}\frac{1-b_0}{1-a_0}\right) + N \log\left(\frac{1-a_1}{1-b_1}\frac{1-b_0}{1-a_0}\right) \geq \log B, \qquad (1.6)$$

and we accept $\mathcal{H}_0$ if

$$S_{1,N} \log\left(\frac{a_1}{b_1}\frac{1-b_1}{1-a_1}\right) - S_{0,N} \log\left(\frac{a_0}{b_0}\frac{1-b_0}{1-a_0}\right) + N \log\left(\frac{1-a_1}{1-b_1}\frac{1-b_0}{1-a_0}\right) \leq \log A.$$

Siegmund (1985) shows that

$$\alpha = P_0\left(l_N \geq B\right) \leq B^{-1}\left(1-\beta\right) \tag{1.7}$$

and

$$\beta = P_1\left(l_N \leq A\right) \leq A\left(1-\alpha\right). \tag{1.8}$$

Hence, given operator defined tolerances α and $\beta,$ by setting

$$B = \frac{1-\beta}{\alpha}, \text{ and } A = \frac{\beta}{1-\alpha}, \tag{1.9}$$

we are guaranteed to satisfy the error probability constraints.

Figure 4 shows two possible sample paths for $SP2$. Again, the continuation region is the area between the two dashed lines representing the bounds. A path exiting the bound corresponding to $\log(B)$ indicates a belief of target presence in AC_1, whereas an exit via the $\log(A)$ bound indicates belief of target presence in AC_0.

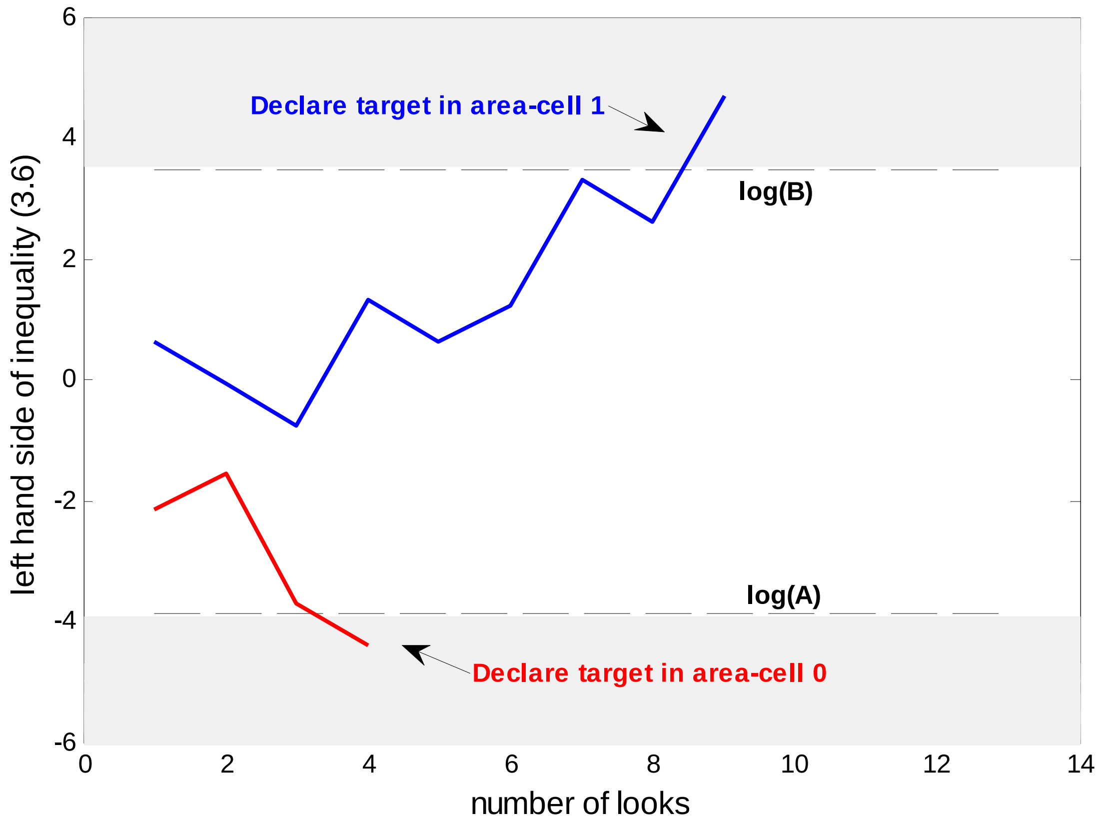

Figure 4. Example sample paths for the two-cell *SST* model case.

3. Case 3: Single Target in One of *m*>2 Area-cells

<u>Procedure *SP*3</u>

Consider an AOI consisting of $m > 2$ area-cells, and a single TOI located within the AOI. Assume, without loss of generality, that the target is hidden in AC_1. The operator can specify the error tolerance in many ways in this case; for example, "If the TOI is in AC_1, I want the probability of saying that the TOI is in AC_2 to be less than 3%, and the probability of saying the TOI is in AC_3 to be less than 2%," etc. For simplicity, in this thesis, consider that the only possible error in this situation is to conclude that the target is not in AC_1. We call such event *ICD* (for *incorrect determination*), and let $P\left(ICD\right) \le \alpha$, for some $\alpha \in \left(0,1\right)$ pre-specified by the operator.

Let C be the set of candidate area-cells; initially all ACs are candidates to contain the target, so that $C = \left\{1,...,m\right\}$. Let $\ell_{(1),n}$ be the largest likelihood ratio at stage n. The idea is to sequentially drop an area-cell from consideration when there is sufficient evidence that it does not contain the target, i.e., when we are confident to within our error tolerance of saying that the TOI is not in that particular area-cell. This suggests eliminating AC_i when $\ell_{(1),n} / \ell_{i,n} \ge B,$ where B is selected to satisfy the bound $P\left(ICD\right) \le \alpha.$

To be more precise, consider area-cell 1 (which contains the TOI) and an arbitrary area-cell $i \ne 1$. Given thresholds $0 < A < B$, let $N_i = \inf\left\{n \ge 1 : \ell_{i,n} / \ell_{1,n} \notin (A,B)\right\}$ be the first time the odds ratio of area-cells 1 and i exits the interval (A,B). Following Equations (1.7) through (1.9), with $B = A^{-1} = (m-1)/\alpha - 1$,we can guarantee the error bound $P\left(\ell_{i,N_i} / \ell_{1,N_i} \ge B\right) \le \alpha / (m-1)$. By Bonferroni's inequality, it follows that

$$P\left(ICD\right) = P\left(\bigcup_{i=2}^{m} \ell_{i,N_i} / \ell_{1,N_i} \ge B\right) \le \sum_{i=2}^{m} P\left(\ell_{i,N_i} / \ell_{1,N_i} \ge B\right) \le \sum_{i=2}^{m} \frac{\alpha}{m-1} = \alpha.$$

Rather than pair-wise comparing all area-cells, it suffices to drop from consideration any area-cell *i* for which $\ell_{(1),n} / \ell_{i,n} \ge (m-1)/\alpha - 1$.

The algorithm proceeds as follows:

<u>Algorithm *SE*</u>

1. Obtain one signal (*sample*) from all area-cells $i \in C.$
2. Compute $\ell_{i,n+1}$ and the ratios $\ell_{(1),n+1} / \ell_{i,n+1}, \forall i \in C.$
3. If $\ell_{(1),n+1} / \ell_{i,n+1} \ge \left(m-1\right)/\alpha - 1,$ then remove *i* from $C.$

4. If $|C|=1,$ stop and declare the single AC in C the *determined* area-cell. Otherwise, increase $n \leftarrow n+1,$ and go back to 1.

The *SE* algorithm is guaranteed to meet the operator-defined tolerance α for the probability of incorrect determination.

IV. COMPUTATIONAL STUDY

This chapter presents results obtained by numerical experimentation in *MATLAB*. We begin our discussion with a description of a notional operational scenario to frame the *SMT* model. We then explain the implementation of the stochastic approximation algorithm presented in Chapter III in the context of this scenario. Finally, we discuss the results of the numerical experiments, and present an analysis of stochastic approximation algorithm performance. We then repeat the process for the case of the *SST* model and sequential eliminating procedure.

A. *SMT* MODEL

1. Scenario Development

For the purposes of our study, we place our notional area of operational interest (AOI) in a 39 square kilometer section of downtown Baghdad, Iraq. Although the *SMT* model does not require that area-cells be uniform (in size or shape) or geographically adjacent, we partition our AOI in this manner as a matter of illustrative and computational convenience. Thus, we begin by discretizing the AOI into uniform area-cells of size 500 by 500 meters. Figure 5 depicts the resulting AOI, consisting of 156 area-cells in a 13 by 12 rectangular grid.

Figure 5. Discretized notional scenario Area of Operational Interest.

At the beginning of the scenario, a single, randomly moving high value target of interest (TOI) is located somewhere within the AOI. In our case, suppose the TOI is a medium-sized white sport utility vehicle known to be an insurgent weapons cache. An intelligence agency has recorded the TOI's hyperspectral signature and provided the library data to the searcher. We construct the initial TOI probability distribution by starting with a *Threat Map* (Riese, 2006), which we aggregate and normalize to produce a probability map that is compatible with our AOI, and whose probabilities sum to one. Table 1 summarizes the initial probability map, with probabilities to three decimal places.

We assume the TOI moves in a Markovian fashion, with a transition matrix that is unknown to the searcher. Appendix C shows the non-zero columns of the ground-truth transition matrix used in our scenario. Table 2 depicts the true steady-state distribution of TOI location (again, unknown to the searcher) resulting from this transition matrix, to three decimal places. For purposes of the numerical experiment, we assume that the target has already reached steady-state.

The searcher possesses a UAS-borne hyperspectral sensor, and a certain number ℓ of available *looks*, or *search budget.* In order to preserve this thesis as unclassified, we assign reasonable random values of hyperspectral sensor sensitivity and specificity against a known signature. Specifically, we assign a random uniform value between 0.75 and 0.99 for sensitivity and a random uniform value between 0.89 and 0.99 for specificity to each area-cell, as depicted to two decimal places in Table 3. It is important to note that, despite our somewhat cavalier method of assigning these numbers, determining the appropriate values operationally can be highly complex (as discussed in Chapter II), and is outside the scope of this thesis. It is left as an option for further study to account for noise in the measured values of sensitivity and specificity.

1 0.007	2 0.011	3 0.018	4 0.016	5 0.011	6 0.013	7 0.015	8 0.005	9 0.007	10 0.005	11 0.004	12 0.005
13 0.004	14 0.002	15 0.004	16 0.005	17 0.007	18 0.007	19 0.013	20 0.004	21 0.004	22 0.005	23 0.004	24 0.005
25 0.004	26 0.004	27 0.004	28 0.004	29 0.007	30 0.009	31 0.009	32 0.011	33 0.004	34 0.002	35 0.004	36 0.004
37 0.004	38 0.005	39 0.007	40 0.007	41 0.004	42 0.009	43 0.013	44 0.009	45 0.011	46 0.007	47 0.011	48 0.007
49 0.002	50 0.002	51 0.007	52 0.011	53 0.015	54 0.005	55 0.018	56 0.015	57 0.016	58 0.009	59 0.009	60 0.004
61 0.002	62 0.002	63 0.004	64 0.011	65 0.009	66 0.009	67 0.004	68 0.004	69 0.009	70 0.009	71 0.005	72 0.007
73 0.004	74 0.002	75 0.002	76 0.005	77 0.005	78 0.009	79 0.007	80 0.007	81 0.009	82 0.009	83 0.007	84 0.005
85 0.005	86 0.004	87 0.002	88 0.005	89 0.007	90 0.009	91 0.015	92 0.005	93 0.002	94 0.005	95 0.005	96 0.009
97 0.007	98 0.002	99 0.002	100 0.004	101 0.007	102 0.007	103 0.004	104 0.007	105 0.007	106 0.004	107 0.007	108 0.007
109 0.007	110 0.007	111 0.004	112 0.004	113 0.002	114 0.004	115 0.005	116 0.004	117 0.007	118 0.002	119 0.005	120 0.007
121 0.009	122 0.007	123 0.009	124 0.002	125 0.004	126 0.005	127 0.007	128 0.004	129 0.004	130 0.002	131 0.005	132 0.005
133 0.015	134 0.013	135 0.004	136 0.004	137 0.005	138 0.005	139 0.009	140 0.004	141 0.002	142 0.002	143 0.005	144 0.004
145 0.005	146 0.004	147 0.002	148 0.002	149 0.004	150 0.005	151 0.007	152 0.007	153 0.009	154 0.009	155 0.005	156 0.004

Table 1. Initial TOI probability map, derived by normalizing the aggregate *Threat Map* (After Riese, 2008).

1 0	2 0	3 0	4 0	5 0	6 0	7 0	8 0	9 0	10 0	11 0	12 0
13 0	14 0	15 0	16 0	17 0	18 0	19 0	20 0	21 0	22 0	23 0	24 0
25 0	26 0	27 0	28 0	29 0	30 0	31 0	32 0	33 0	34 0	35 0	36 0
37 0	38 0	39 0	40 0	41 0	42 0	43 0	44 0	45 0	46 0	47 0	48 0
49 0	50 0	51 0	52 0	53 0	54 0	55 0	56 0	57 0	58 0.021	59 0	60 0
61 0	62 0	63 0	64 0	65 0	66 0	67 0	68 0	69 0	70 0.021	71 0.025	72 0
73 0	74 0	75 0	76 0.03	77 0.033	78 0.056	79 0	80 0	81 0	82 0	83 0	84 0
85 0	86 0	87 0	88 0.015	89 0.046	90 0.028	91 0	92 0	93 0	94 0	95 0	96 0
97 0	98 0	99 0	100 0.038	101 0.053	102 0.062	103 0.088	104 0.085	105 0	106 0	107 0	108 0
109 0	110 0	111 0	112 0.097	113 0.103	114 0.199	115 0	116 0	117 0	118 0	119 0	120 0
121 0	122 0	123 0	124 0	125 0	126 0	127 0	128 0	129 0	130 0	131 0	132 0
133 0	134 0	135 0	136 0	137 0	138 0	139 0	140 0	141 0	142 0	143 0	144 0
145 0	146 0	147 0	148 0	149 0	150 0	151 0	152 0	153 0	154 0	155 0	156 0

Table 2. Notional TOI steady-state distribution.

1 0.764 0.907	2 0.78 0.985	3 0.763 0.918	4 0.83 0.902	5 0.783 0.915	6 0.862 0.914	7 0.794 0.972	8 0.779 0.909	9 0.981 0.934	10 0.768 0.91	11 0.883 0.934	12 0.803 0.937
13 0.932 0.896	14 0.858 0.93	15 0.861 0.986	16 0.902 0.909	17 0.833 0.937	18 0.922 0.906	19 0.859 0.892	20 0.858 0.901	21 0.894 0.932	22 0.761 0.906	23 0.779 0.973	24 0.925 0.935
25 0.828 0.907	26 0.873 0.918	27 0.768 0.982	28 0.852 0.961	29 0.81 0.976	30 0.772 0.908	31 0.903 0.92	32 0.884 0.964	33 0.879 0.987	34 0.906 0.904	35 0.962 0.899	36 0.833 0.955
37 0.79 0.891	38 0.945 0.909	39 0.946 0.905	40 0.93 0.9	41 0.945 0.965	42 0.895 0.973	43 0.818 0.966	44 0.827 0.892	45 0.914 0.951	46 0.827 0.987	47 0.953 0.915	48 0.877 0.97
49 0.885 0.963	50 0.987 0.942	51 0.953 0.935	52 0.906 0.942	53 0.77 0.975	54 0.963 0.901	55 0.93 0.945	56 0.852 0.943	57 0.831 0.979	58 0.952 0.939	59 0.923 0.984	60 0.837 0.978
61 0.954 0.969	62 0.752 0.919	63 0.755 0.978	64 0.756 0.984	65 0.889 0.988	66 0.828 0.971	67 0.799 0.93	68 0.857 0.982	69 0.821 0.903	70 0.93 0.902	71 0.929 0.926	72 0.869 0.935
73 0.863 0.956	74 0.874 0.919	75 0.797 0.911	76 0.984 0.983	77 0.765 0.905	78 0.904 0.979	79 0.812 0.93	80 0.751 0.895	81 0.932 0.909	82 0.848 0.897	83 0.812 0.974	84 0.818 0.915
85 0.771 0.896	86 0.782 0.937	87 0.766 0.932	88 0.846 0.941	89 0.965 0.939	90 0.771 0.911	91 0.829 0.981	92 0.834 0.899	93 0.834 0.901	94 0.919 0.949	95 0.892 0.915	96 0.881 0.973
97 0.768 0.959	98 0.804 0.926	99 0.76 0.966	100 0.945 0.987	101 0.84 0.959	102 0.966 0.918	103 0.887 0.892	104 0.855 0.932	105 0.822 0.986	106 0.94 0.983	107 0.824 0.938	108 0.795 0.962
109 0.979 0.921	110 0.916 0.939	111 0.791 0.919	112 0.768 0.99	113 0.783 0.942	114 0.762 0.898	115 0.81 0.918	116 0.915 0.964	117 0.861 0.986	118 0.755 0.904	119 0.848 0.902	120 0.884 0.932
121 0.936 0.9	122 0.848 0.935	123 0.774 0.944	124 0.935 0.96	125 0.987 0.938	126 0.864 0.988	127 0.914 0.988	128 0.869 0.898	129 0.944 0.894	130 0.792 0.916	131 0.827 0.909	132 0.943 0.919
133 0.832 0.987	134 0.942 0.909	135 0.941 0.976	136 0.966 0.908	137 0.973 0.925	138 0.884 0.979	139 0.967 0.897	140 0.944 0.979	141 0.887 0.922	142 0.768 0.985	143 0.927 0.979	144 0.907 0.984
145 0.864 0.895	146 0.759 0.898	147 0.909 0.957	148 0.906 0.989	149 0.817 0.985	150 0.869 0.929	151 0.778 0.964	152 0.801 0.946	153 0.886 0.934	154 0.811 0.934	155 0.803 0.894	156 0.785 0.985

Table 3. Notional hyperspectral sensor *sensitivity* (center cell values) and *specificity* (bottom cell values).

In each time step, the searcher allocates one look in a chosen area-cell, and the search budget is subsequently decremented. Upon each look, the sensor returns a signal of *hot* if it senses detection (of course, correct detections and false positives are indistinguishable to the sensor), and a signal of *cold* otherwise. The scenario then advances forward one time step, and the process continues until the entire search budget is exhausted. Recall that the objective for the searcher is to determine the steady-state location distribution of the TOI as quickly as possible (to within the absolute error tolerance), based on the signals from the sensor and knowledge of its sensitivity and specificity.

2. Stochastic Approximation Algorithm Implementation and Results

We implement our model in the framework of the described notional operational scenario, using *MATLAB*. Sample *MATLAB* code for the *SA* algorithm may be found in Appendix A.

In order to evaluate the *SA* algorithm's ability to approximate the true steady-state distribution of the target and the associated near-optimal sampling rates, one replication of 50,000 iterations was performed on the 156-cell AOI. Table 4 depicts the algorithm's cell sampling rates compared to the theoretical optimal area-cell sampling rates which, as discussed in Chapter III, depend not only on the absolute error tolerance (ε), sensor sensitivity (a_i), and specificity $(1-b_i)$ values, but also on the true steady-state distribution (π_i). Cell sampling rates are given to four decimal places.

1	2	3	4	5	6	7	8	9	10	11	12
0.0001	0	0	0.0002	0.0001	0.0004	0	0.0001	0.0004	0.0001	0.0003	0
0.0001	0	0	0	0.0001	0.0002	0	0.0001	0.0004	0.0002	0.0001	0.0002
13	14	15	16	17	18	19	20	21	22	23	24
0.0011	0.0002	0	0.0213	0.0001	0.0053	0.0013	0.0007	0.0007	0.0001	0	0.0097
0.0003	0.0001	0	0.0251	0.0001	0.0071	0.001	0.0002	0.0002	0	0	0.0097
25	26	27	28	29	30	31	32	33	34	35	36
0.0002	0.0006	0	0	0	0.0001	0.004	0.0001	0	0.4014	0.0003	0
0.0001	0.001	0	0.0001	0	0	0.0043	0.0003	0	0.4394	0.0003	0.0002
37	38	39	40	41	42	43	44	45	46	47	48
0.0001	0.0009	0.0008	0.0014	0.0013	0.0001	0	0.0003	0.0005	0	0.0008	0
0	0.0014	0.0002	0.0017	0.0002	0	0	0.0001	0.0001	0	0.0003	0
49	50	51	52	53	54	55	56	57	58	59	60
0.0001	0.0004	0.0029	0.0006	0	0.0004	0.0038	0.0001	0	0.0056	0.0001	0
0	0.0001	0.0038	0.0001	0	0.0006	0.0004	0.0001	0	0.0004	0	0
61	62	63	64	65	66	67	68	69	70	71	72
0.0019	0	0	0	0	0	0.0001	0	0.0002	0.0017	0.122	0.0002
0.0003	0	0	0	0	0	0	0.0001	0.0001	0.0017	0.1322	0.0001
73	74	75	76	77	78	79	80	81	82	83	84
0.0001	0.0006	0.0001	0.3063	0.0001	0.0001	0.0001	0.0001	0.0022	0.0006	0	0.0001
0.0001	0.0007	0.0001	0.2655	0	0	0	0.0001	0.0019	0.0001	0.0001	0.0001
85	86	87	88	89	90	91	92	93	94	95	96
0.0001	0	0	0.0001	0.0012	0.0001	0	0.0003	0.0003	0.0008	0.0022	0
0.0001	0	0	0	0.0014	0	0	0.0001	0.0002	0.0001	0.0003	0
97	98	99	100	101	102	103	104	105	106	107	108
0	0.0001	0	0.0001	0	0.0005	0.0709	0.0002	0	0.0001	0.0001	0
0	0.0002	0	0.0001	0	0.0013	0.0792	0.0004	0	0.0002	0.0001	0
109	110	111	112	113	114	115	116	117	118	119	120
0.0003	0.0016	0.0001	0	0	0.0001	0.0001	0.0002	0	0.0001	0.0005	0.0004
0.0004	0.0017	0	0	0.0001	0.0001	0	0.0001	0	0	0.0001	0.001
121	122	123	124	125	126	127	128	129	130	131	132
0.0011	0.0001	0	0.0009	0.0004	0	0	0.0015	0.0006	0.0001	0.0002	0.0019
0.0015	0.0001	0	0.0009	0.0001	0	0	0.0003	0.0006	0	0.0005	0.0003
133	134	135	136	137	138	139	140	141	142	143	144
0	0.0011	0.0003	0.0004	0.0004	0	0.0003	0.0002	0.0009	0	0.0001	0
0	0.0012	0.0001	0.0001	0.0007	0	0.0001	0.0007	0.0007	0	0.0001	0.0001
145	146	147	148	149	150	151	152	153	154	156	156
0.0014	0.0001	0.0003	0	0	0.0003	0	0	0.0004	0.0001	0.0002	0
0.0002	0	0.0001	0	0	0.0001	0	0	0.0001	0	0.0001	0

Table 4. Theoretical optimal search frequencies (center cell values) and *SA* algorithm resultant search frequencies (bottom cell values, based on 50,000 iterations).

It is difficult to infer a degree of model success based upon these values, beyond the fact that the *SA* algorithm provides sampling rates that appear to be on the same order of magnitude as the theoretical optimal sampling rates. It is therefore more useful to evaluate the model based upon its ability to estimate the TOI's steady-state distribution to within a certain absolute error tolerance, which we provide below in Section 3.

Figure 6 depicts estimates of the true steady-state target position distributions for three select area-cells. The estimates provided by the *SA* algorithm are compared to the estimates generated by a uniform random search. Both estimates appear to converge upon the true steady-state distribution as the total number of looks becomes large, in accordance with the Strong Law of Large Numbers. One might conjecture from this figure that the total error—depicted by the aggregate area between each estimate and the true steady state after the transient has worn off—is less for the *SA* estimate than for the random uniform. However, it is difficult to say whether this is in fact the case, or just an artifact of this particular numerical experiment, as this is only the figure only shows one replication's data for three area-cells. This further suggests that a more useful measure of model effectiveness will be the error decay rate, again provided in Section 3.

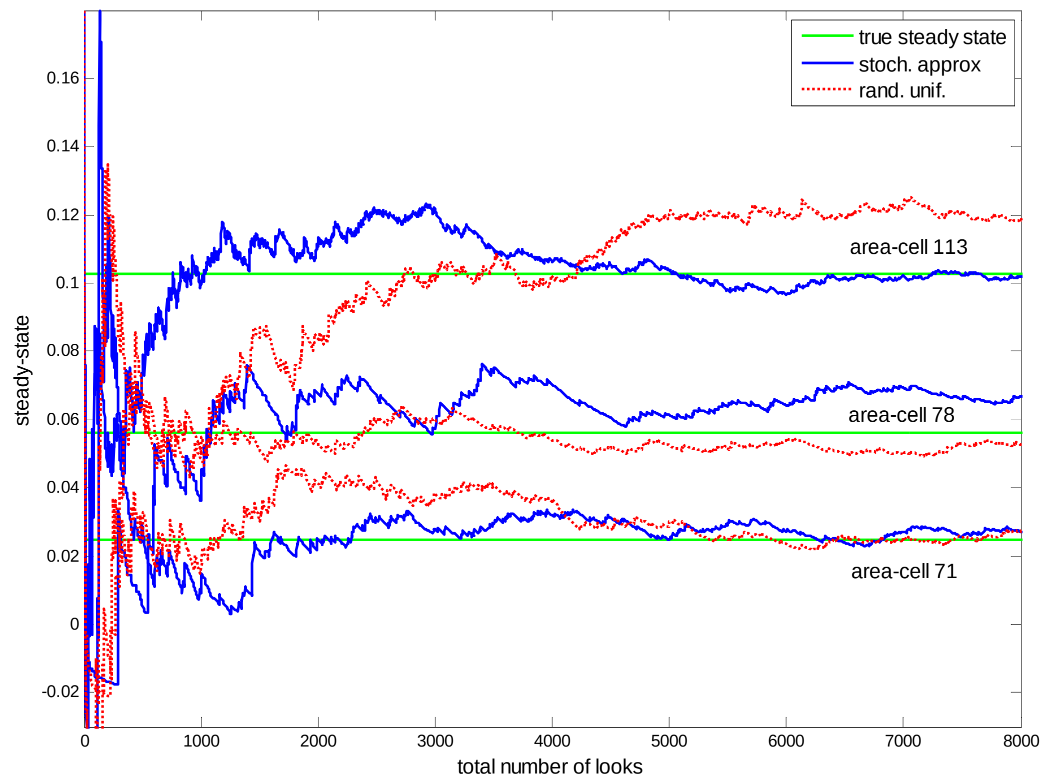

Figure 6. Number of looks versus steady-state estimate for one sample path of three particular area-cells.

In order to evaluate the absolute error rate, we count the number of replications for which the algorithm's estimate of the steady-state distribution is outside of some pre-defined absolute tolerance (in our case, an absolute error tolerance of 0.05 was used), for search budgets at each integer level between 1 and 6500, and divide by the total number of replications. Due physical memory and time constraints (a single replication on the 156-cell AOI took over two hours to compute), the absolute error decay rate was evaluated by conducting 10,000 replications with a simplified AOI of only four area-cells (results are depicted in Figures 7 and 8). However, the error rate results generalize to problems of any size, as shown in Chapter III.

3. Analysis of Stochastic Approximation Algorithm Performance

Recall that the measure of effectiveness in this model is to minimize the probability that the resultant absolute error in the estimate of steady-state distribution is greater than some tolerance ε. In Chapter III, we acknowledged that any search plan that allocates a positive fraction of the search budget to all cells would lead to an absolute error rate that decays to zero exponentially as the search budget goes to infinity. Figure 6 supported this by showing that the *SA* algorithm estimate and a random uniform estimate both converge to the true steady-state values.

Additionally, we claimed that the search frequencies determined by the *SA* algorithm exhibit error rates that decay at the fastest rate possible. Figure 7 depicts the average resultant absolute error decay rates (over 10,000 sample paths of 6,500 iterations on a simpler, four-cell problem) for four different strategies:

- A random uniform search.
- A constant naïve estimate, where the determination thresholds are the midpoints between a_i and b_i. If the allocations are inversely proportional to the distance $a_i - \frac{(a_i - b_i)}{2}$, the allocation becomes $\hat{p}_i = \frac{(a_i - b_i)^{-1}}{\sum_{j=1}^{m}(a_j - b_j)^{-1}}$.
- The stochastic approximation algorithm search discussed in this section.
- A hypothetical optimal search plan in which the searcher knows *a priori* the theoretical optimal search frequencies derived in Chapter III.

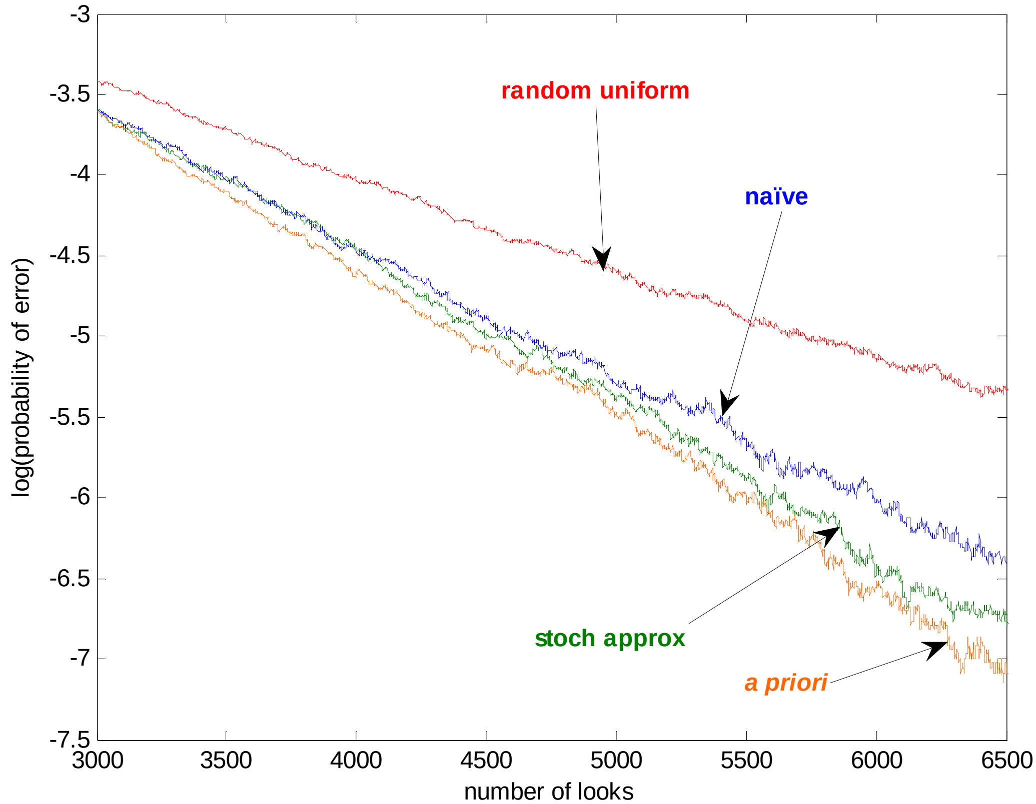

Figure 7. Large number of looks versus the logarithm of probability of error.

As predicted, the *a priori* plan appears to provide a bound on error decay rate and as the search budget grows large, the *SA* algorithm outperforms the random uniform search. The naïve allocation performs relatively well, with an error rate tending more toward the optimal than toward the random (see Figure 7).

One important feature of the model deserves discussion. Recall that a limitation of the *SMT* model is that it applies only to situations involving large search budgets. Note that, for a relatively small search budget (say, less than about 1000 in the current example), any efficiency gained by the *SA* algorithm is negligible. In fact, it is often the case that the random uniform error rate decays faster for than that of the *SA* algorithm when the number of looks is small (see Figure 8). Additionally, the naïve allocation performs nearly as efficiently as the random in this situation.

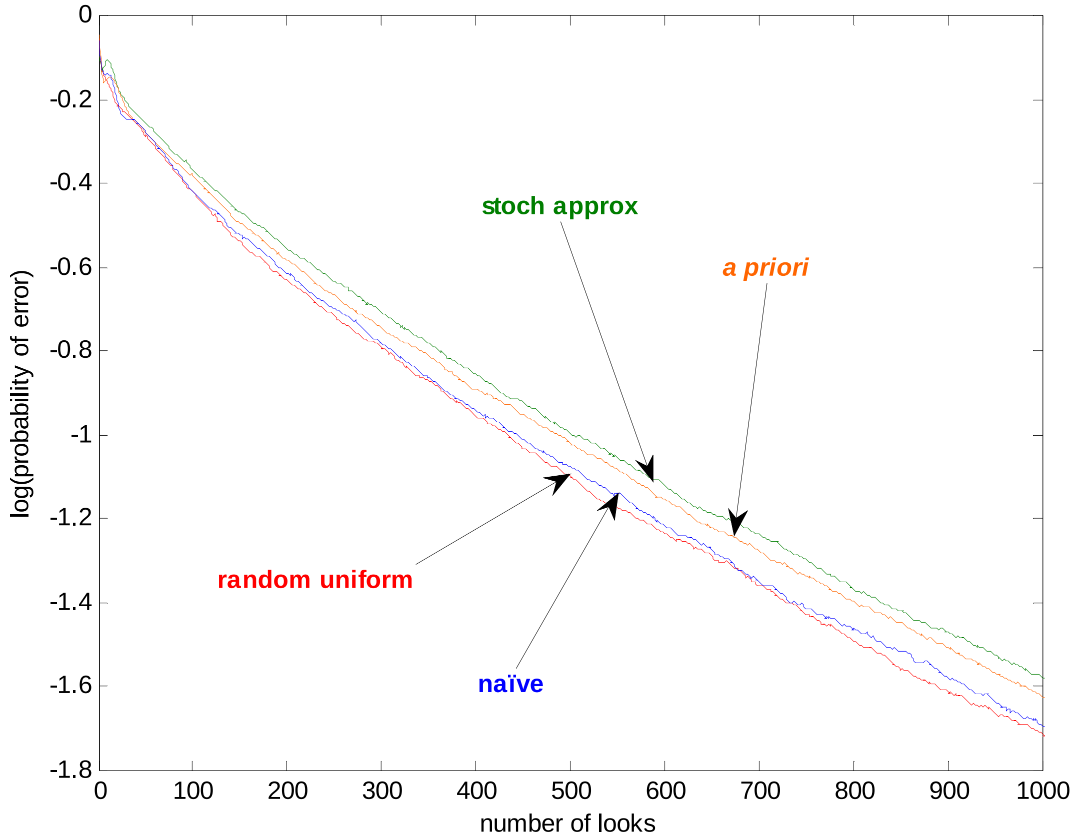

Figure 8. Small number of looks versus the logarithm of probability of error.

B. *SST* MODEL

1. Scenario Development

Consider the same discretized AOI as for the adaptive case (see Figure 5). Sensor sensitivity and specificity remain as in Table 3. We first consider the case where each area-cell either contains or does not contain a stationary TOI, independent of all other cells, and refer to this as case 1. This allows us to perform the calculations on one area-cell, and generalize the results to all area-cells. We then modify the scenario to the case in which a single stationary TOI is present in only one of two area-cells (case 2). Finally, we generalize to the case of $m > 2$ area-cells (case 3).

2. Sequential Eliminating Procedure Implementation and Results

As with the *SMT* model, we choose *MATLAB* as a computational tool for the numerical experiments for the *SST* model. Sample *MATLAB* code for the sequential eliminating procedure may be found in Appendix B.

For case 1, an arbitrary area-cell containing a TOI was chosen. To observe the effect of decreasing the difference between sensor sensitivity and (1-specificity) on the expected number of looks, b was held constant at 0.35, and a was varied from 0.7 to 0.35 at increments of 0.002. 10,000 replications were conducted at each increment. Predictably, the closer the values of a (sensitivity) and b (1-specificity) are to one another, the larger the number of looks required to make a determination. Figure 9 verifies this intuition; in fact, the increase appears to be polynomial in the complement of the difference.

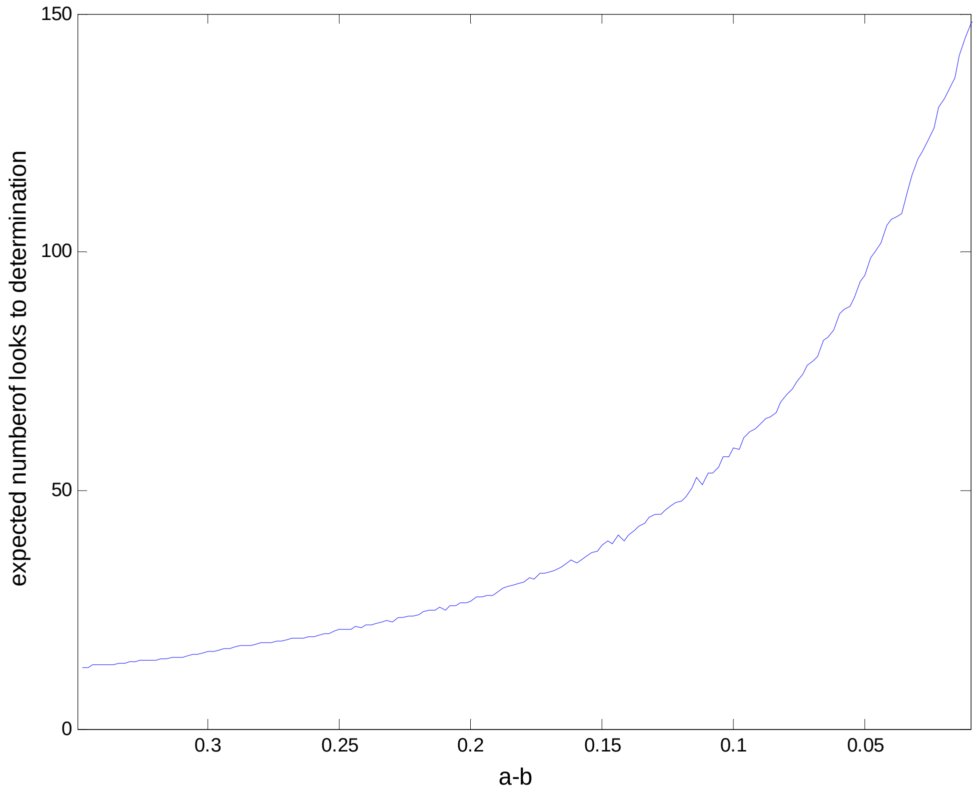

Figure 9. Polynomial increase in number of looks as difference between a and b gets small ($m = 1$, $\alpha = 0.05$).

For cases 2 and 3, we arbitrarily (and without loss of generality) placed the target in AC_1. The error tolerance α remained fixed at 0.05. Starting with $m = 2$ area-cells and working up by adding one AC per iteration until encompassing all 156 area-cells, we performed 50,000 replications at each iteration in order to determine achieved error rates and expected number of looks as m increases. Figure 10 depicts the near-linear increase in observed number of looks as m increases for case 3 with fixed $\alpha = 0.05$.

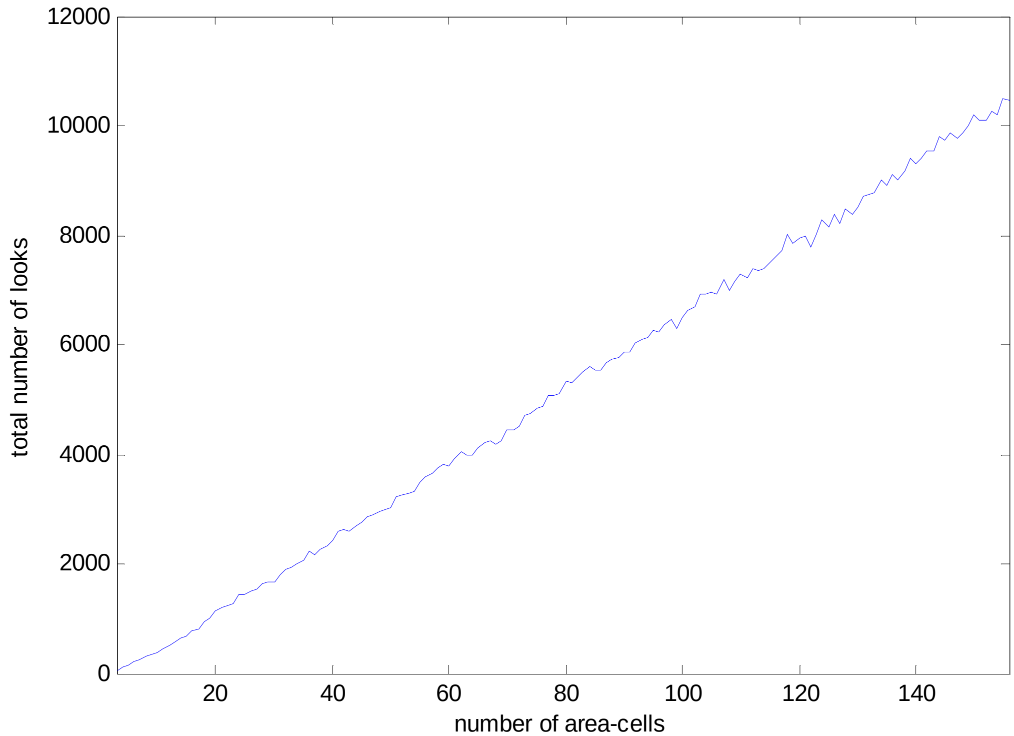

Figure 10. Near-linear relationship between m and number of looks, case 3, $\alpha = 0.05$.

To determine the effect of varying error tolerance, we chose to fix $m = 10$. The type-I error probability threshold α was varied between 0.01 and 0.1 at 0.001 increments. At each level of α, 50,000 replications were performed to calculate the observed miss rate and average number of looks until a determination of target presence or absence was made. Naturally, the number of expected looks decreases with an increase in error tolerance for the sequential procedure. Figure 11 highlights the relationship.

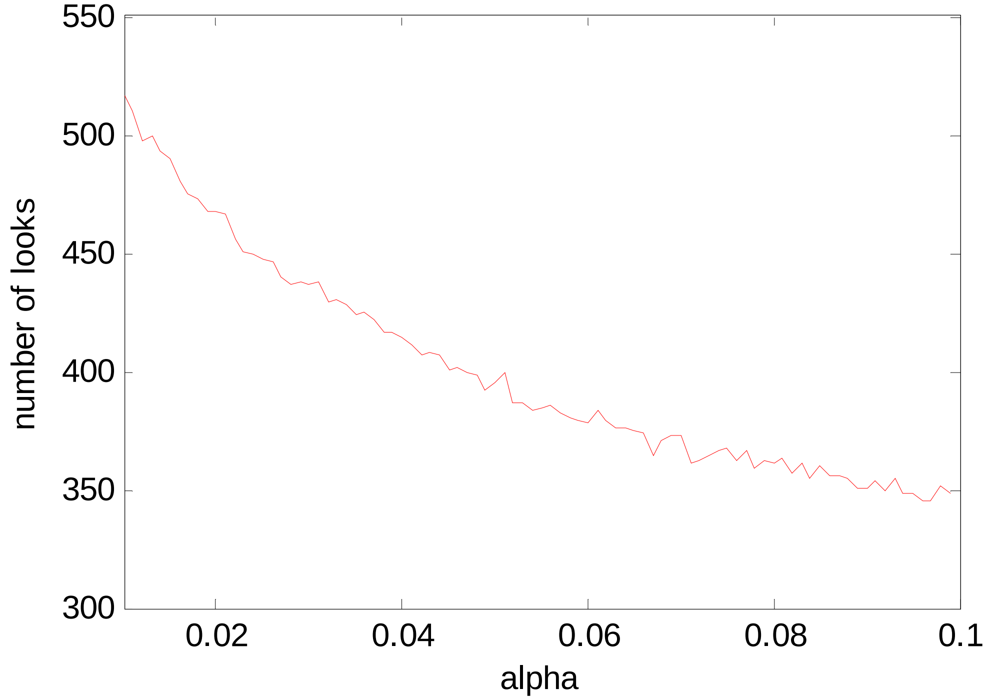

Figure 11. Effect of varying α on expected number of looks (m=10).

3. Analysis of Sequential Eliminating Procedure Performance

Recall that the goal of the sequential eliminating procedure is to indicate target location with an accuracy rate guaranteed to meet operator-specified error tolerances, within a reasonable number of expected looks. One possible measure of performance is the amount of slack between the error tolerance and the observed error rate. Intuitively, the less slack, the fewer number of expected looks would be required. However, if a method with the same expected number of looks as the sequential eliminating procedure exhibits a larger observed error rate, it is reasonable to state that the sequential eliminating procedure is more efficient than such a method. (Alternatively, one could invoke a method exhibiting the same achieved error rate and compare the expected number of looks, but we choose the former scheme for ease of computation and illustration.)

For comparison purposes, consider a naïve allocation method with the same total expected number of looks as the sequential eliminating procedure. Let $\bar{\tau}_m$ be the total average number of looks for the sequential eliminating procedure with an AOI consisting of m area-cells. Let $\overline{\lambda}_m$ be the average number of looks per area-cell for the naïve allocation, which allocates an equal number of looks to each area-cell, and does not eliminate area-cells from contention, so that $\overline{\lambda}_m = \overline{\tau}_m / m$. (Since $\overline{\lambda}_m$ is likely not an integer value, we set

$$\lambda_m = \begin{cases} \lceil \overline{\lambda}_m \rceil & \text{w.p. } p \\ \lfloor \overline{\lambda}_m \rfloor & \text{w.p. } 1\text{-}\ p, \end{cases}$$

where $p = \overline{\lambda}_m - \lfloor \overline{\lambda}_m \rfloor$, in order to approach the desired average over many iterations.) After λ_m looks into each area-cell, the area-cell with the largest likelihood ratio is declared to contain the target for the naïve model. We begin our comparison by examining the effect of varying α with m=10. We then fix α and complete 5,000 iterations at each integer $m = 3,\ldots,156$, comparing the observed error rates of the sequential eliminating procedure with that of the naïve model.

Figure 12 shows that, with fixed a, b and β for the multiple-cell case, an increase in α appears to affect an increase in the observed miss rate in a manner that preserves the *ratio* of α and the observed rate for the sequential eliminating procedure. Also note that, for the same average number of looks at each value of α, the observed error rate for the naïve method is considerably larger for all values of α (indeed, the naïve rates exceed the threshold for all values of α).

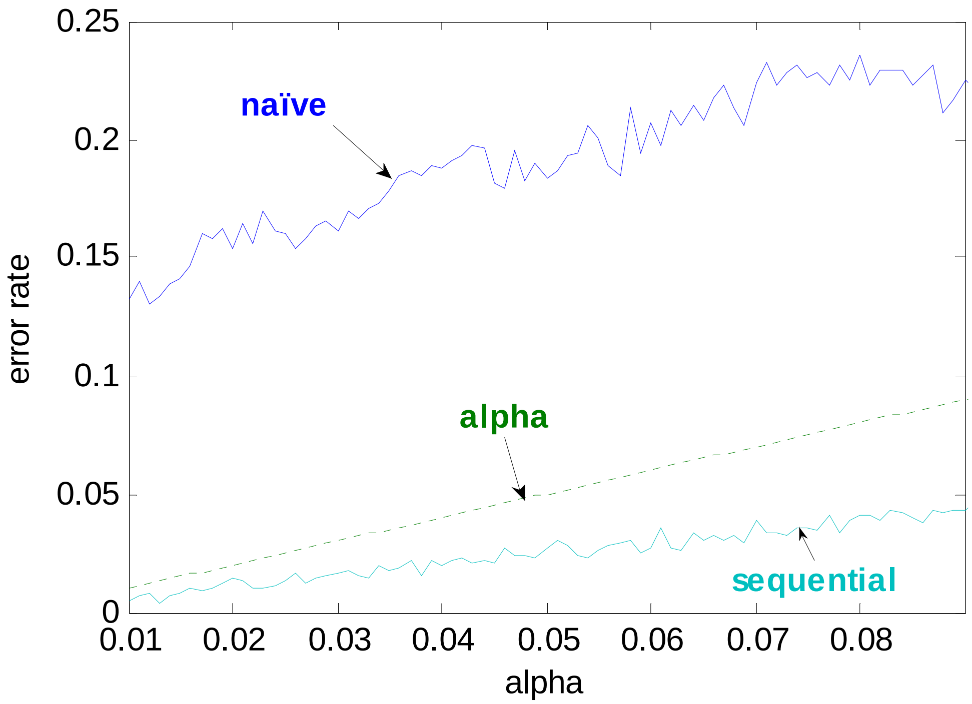

Figure 12. Effect of varying α on observed error rates (m=10).

Figure 13 illustrates how the observed error rate for the sequential eliminating procedure appears to converge as the number of area-cells becomes large. The observed rate is arguably reasonable when compared to the threshold. As with the case of varying α, the error rate for the naïve method is considerably higher than that of the sequential eliminating procedure for the same total number of looks for all values of m. Indeed, the naïve error rate is once again consistently above the threshold.

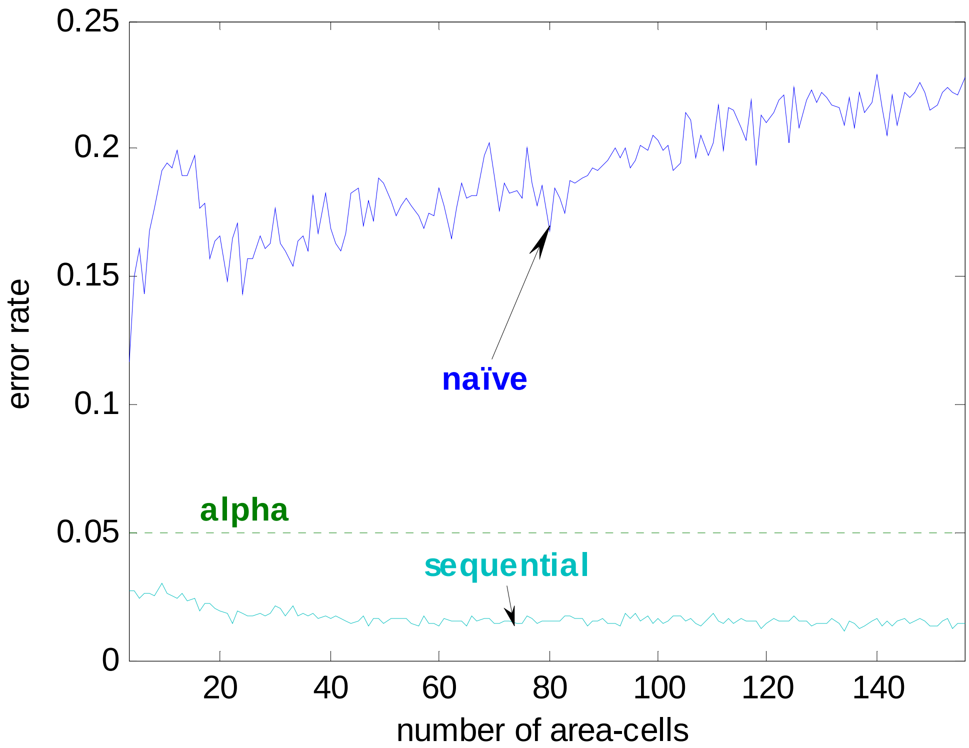

Figure 13. Achieved error rates with a threshold of 0.05, based on 5,000 iterations at each integer level of 3 to 156 cells.

THIS PAGE INTENTIONALLY LEFT BLANK

V. CONCLUSIONS AND FUTURE WORK

This chapter summarizes the findings of the computational study, as well as offers some possible areas for future study, for both of the models presented in this thesis.

A. CONCLUSIONS

1. Single Markov Target Model

Overall, we conclude that the stochastic approximation algorithm provides sufficiently improved solutions over all other strategies for large search budgets. While this may not be significant on the tactical level (which is often characterized by a constrained search budget), it is certainly appropriate on the operational-strategic level wherein one is concerned with long-term Intelligence, Surveillance, and Reconnaissance operations and Target of Interest pattern recognition. Additionally noteworthy is that, for circumstances in which the stochastic approximation algorithm is unavailable, the naïve estimate provides sufficiently improved solutions over the uniform random strategy for large search budgets, and better solutions than the stochastic approximation strategy for small search budgets.

2. Single Static Target Model

The sequential eliminating procedures presented in this thesis provide efficient results guaranteed to meet desired error rate thresholds for a variety of scenarios. While the gap between the error threshold and the achieved error rates might conceivably be tightened, any efficiency gained in terms of reducing the number of looks required is likely not operationally significant when compared with the risk of violating the threshold. We showed that a naïve, non-eliminating method not only demonstrates consistently higher error rates for the same total number of looks for all appropriate values of both number of area-cells and error tolerance; there is also no guarantee that the naïve procedure's results meet the operator-prescribed error tolerance. It is therefore reasonable to conclude that the sequential eliminating procedure is more efficient than the naïve allocation method.

B. AREAS FOR POSSIBLE FUTURE STUDY

1. Single Markov Target Model

As referred to throughout this thesis, there are several areas deserving further study regarding our Single Markov Target model that we were simply unable to treat here due to time and scope limitations. These include, for example:

- Study the effects of pre-steady-state target Markovian movements.
- Study the effects non-Markovian movements.
- Study the effect of counter-detection; say, a target of interest becomes alerted to the presence of the sensor with a certain probability following a look into the area-cell containing the target of interest, and subsequently moves according to a different transition matrix.
- Allow for spatial and/or temporal correlation among sensor observations.
- Consider noise within the observed measurements of sensor sensitivity and specificity.
- Determine breakpoint criteria for when the adaptive algorithm is advantageous (i.e., what constitutes a "large" search budget).
- Consider multiple sensors and/or multiple targets.

2. Single Static Target Model

As with the Single Markov Target model, areas for possible further study include allowing for multiple targets and/or sensors, and modeling sensor sensitivity and specificity stochastically. It may be of interest to model counter-detection by, say, decreasing the associated value of sensor sensitivity with a certain probability following a look into the area-cell containing the target of interest. Additionally, a formal sensitivity analysis is recommended in order to determine any operational significant efficiency gained by tightening the slack between observed error rate and the error tolerance in order to decrease the expected number of looks. Finally, it may be worthwhile to investigate other ways in which to specify error tolerances for the multiple-cell case such as allowing the error tolerance to vary by area-cell.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A

The following is *MATLAB* code used for the adaptive algorithm, using a four-cell example problem.

```
function bos=markov()
iter=7000;% number of iterations
reps=7000
ss=[0.1 0.4 0.3 0.2];%target steady state distn (unknown to searcher)
a=[0.8 0.94 0.7 0.95];%sensitivity
b=[0.1 0.04 0.15 0.05];%1-specificity
prior = [.4, .1, .2, .3];%target pmf, (known to searcher)
mu=a.*(ss)+b.*(1-ss);%true mu, the value to which # of detects will
converge
dim=size(ss,2); % # area-cells
epsilon=0.05; %absolute error tolerance
lb=b+(a-b).*(ss-epsilon) % lower bound of adequate allocation
ub=b+(a-b).*(ss+epsilon)% upper bound of adequate allocation
iub=ub.*log(ub./mu)+(1-ub).*log((1-ub)./(1-mu));%ld rate function of ub
ilb=lb.*log(lb./mu)+(1-lb).*log((1-lb)./(1-mu));%ld rate function of lb
truedist=min(ilb,iub); % true ld rate function
truefrequencies=(1./truedist)./sum(1./truedist);%true optimal search
frequencies
nfreqs=(1./(a-b))./sum(1./(a-b));%naive search frequencies
for k=1:reps
    x=b+(a-b).*prior;% initializes x for adaptive based on prior pmf
    xr=x;% initializes x for random
    xt=x;% initializes x for a priori
    xn=x;
    gub=x+(a-b)*epsilon;
    glb=x-(a-b)*epsilon;
    ixub=gub.*log(gub./x)+(1-gub).*log((1-gub)./(1-x));%ld rate
function of ub
    ixlb=glb.*log(glb./x)+(1-glb).*log((1-glb)./(1-x));%ld rate
function of ub
    dist=min(ixub,ixlb); %ld rate function of min
    bucket=dist.^(-1)/sum(dist.^(-1));%pmf for ld rate function
    p5=[0, truefrequencies(1:dim-1)]; %interval setup for a priori
    p6=truefrequencies(1:dim); %interval setup for a priori
    p7=[0, nfreqs(1:dim-1)]; %interval setup for naive
    p8=nfreqs(1:dim); %interval setup for a naive

    s=ones(1,dim); %init sample sizes to one for each cell
    sr=s;%init sample sizes to one for each cell
    st=s;%init sample sizes to one for each cell
    sn=s; %init sample sizes to one for each cell

    for i=1:iter
        p1=[0, bucket(1:dim-1)];%interval setup for adaptive
        p2=bucket(1:dim);%interval setup for adaptive
        u=rand;%rand for xi
        u3=rand;%rand for a priori index
```

```
        index=sum((1:dim).*(cumsum(p1)<=u).*(u<cumsum(p2)));%xi
        indexr=unidrnd(4);%randint(1,1,[1,4]);%random uniform cell
        indext=sum((1:dim).*(cumsum(p5)<=u3).*(u3<cumsum(p6)));%cell
based on a priori optimal allocation
        indexn=sum((1:dim).*(cumsum(p7)<=u3).*(u3<cumsum(p8)));%cell
based on naive est
        s(index)=s(index)+1;%update sample size bernoulli for adaptive
        sr(indexr)=sr(indexr)+1;%update sample size for random
        st(indext)=st(indext)+1;%update sample size for a priori
        sn(indexn)=st(indexn)+1;%update sample size for naive

        r=rand<=mu(index);%detection bernoulli for adaptive
        rr=rand<=mu(indexr);%detection bernoulli for random
        rt=rand<=mu(indext);
        rn=rand<=mu(indexn);

        x(index) = x(index)+(r-x(index))/s(index);%update avg for
adaptive
        xr(indexr) = xr(indexr)+(rr-xr(indexr))/sr(indexr);%update avg
for random
        xt(indext) = xt(indext)+(rt-xt(indext))/st(indext);%update avg
for a priori
        xn(indexn) = xt(indexn)+(rn-xn(indexn))/sn(indexn);%update avg
for a priori

        gub(index)=x(index)+(a(index)-b(index))*epsilon;
        glb(index)=x(index)-(a(index)-b(index))*epsilon;

        ixlb(index)=glb(index).*log(glb(index)./x(index))+(1-
glb(index)).*log((1-glb(index))./(1-x(index)));%ld rate function of lb
        ixub(index)=gub(index).*log(gub(index)./x(index))+(1-
gub(index)).*log((1-gub(index))./(1-x(index)));%ld rate function of ub

        dist=min(ixub,ixlb);%l.d. rate function update
        bucket=dist.^(-1)/sum(dist.^(-1));%pmf for ld rate function
        pihat=(x-b)./(a-b);
        pihatr=(xr-b)./(a-b);
        pihatt=(xt-b)./(a-b);
        pihatn=(xn-b)./(a-b);
        pihats(i,:)=pihat;
        pihatsr(i,:)=pihatr;
        pihatst(i,:)=pihatt;
        pihatsn(i,:)=pihatn;

        wrong(k,i)=mean(abs(pihat-ss)>epsilon);
        wrongr(k,i)=mean(abs(pihatr-ss)>epsilon);
        wrongt1(k,i)=mean(abs(pihatt-ss)>epsilon);
        wrongn(k,i)=mean(abs(pihatn-ss)>epsilon);
        wrongt(k,i)=(exp(-i./sum(1./truedist)));
    end

    k
end
```

```
pwrong=mean(wrong);
pwrongr=mean(wrongr);
pwrongt=mean(wrongt);
pwrongt1=mean(wrongt1);
pwrongn=mean(wrongn);
pihat
%pihatr
%pihatt
ss
truefrequencies
frequencies=(s-1)/iter
truemu=ones(iter,1)*mu;
sstate=ones(iter,1)*ss;
truefreqs=ones(iter,1)*truefrequencies;
%
plot(1:iter,log(pwrong),'b',1:iter,log(pwrongr),'r',1:iter,log(pwrongt)
,'g',1:iter,log(pwrongt1),'c',1:iter,-(1:iter)/sum(1./truedist),'y')
% plot(1:iter,log(pwrong),'b',1:iter,log(pwrongr),'r',1:iter,-
(1:iter)/sum(1./truedist),'g')
%     count
%     countr
%     countt
%     plot(perrort)
%     plot(sstate)
%     avgtgt=tgt./iter

end
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B

The following is *MATLAB* code used for the sequential eliminating procedure, varying the number of area-cells from three to 156, with the TOI located in AC_1.

```
function bos=seq3()
reps=5000;% number of replications

at=[0.764377369 0.78013237  0.763048438 0.829653856 0.783187936
0.861500282 0.7940199   0.778956788 0.980568175 0.767952462 0.882596083
0.802665581 0.932069314 0.857595049 0.861134275 0.901862303 0.832543914
0.921926775 0.858781837 0.857607031 0.894258574 0.760550857 0.77867332
0.92494074  0.827953419 0.872731785 0.767612636 0.851690738 0.809917345
0.771560887 0.902829194 0.883563397 0.879007539 0.906012706 0.9615216
0.833174881 0.789552179 0.945467317 0.946057278 0.930473901 0.945149796
0.895457497 0.818231974 0.827209992 0.913855436 0.826787338 0.952930998
0.876803421 0.885381608 0.987431778 0.953051    0.906448316 0.770497982
0.962819268 0.930193065 0.852155027 0.83147321  0.951727379 0.923018754
0.837080099 0.953565399 0.752180624 0.755233645 0.755896192 0.889144868
0.828107202 0.799289558 0.857286381 0.82134985  0.929926846 0.929295357
0.868960987 0.863498849 0.874244484 0.796966453 0.983966875 0.76516679
0.903890473 0.812371407 0.750526499 0.932096997 0.848402882 0.812296006
0.817516651 0.771200914 0.781980008 0.765662689 0.845692943 0.965458891
0.771303387 0.829035747 0.834196225 0.833510047 0.918708516 0.891839986
0.880859016 0.768251293 0.80372802  0.760343658 0.945212372 0.840359708
0.965729992 0.887229029 0.855403358 0.821667225 0.940151288 0.824287152
0.79544627  0.978719313 0.915907164 0.791414375 0.768399415 0.783205648
0.762326265 0.809792289 0.914770558 0.861269546 0.754924019 0.848123228
0.883559472 0.935584267 0.847992869 0.773887555 0.935168021 0.987139592
0.864397497 0.914056384 0.868620928 0.943567157 0.792172003 0.826948986
0.942816637 0.831512882 0.942243958 0.940664075 0.965565544 0.973488383
0.884480939 0.967052613 0.943980301 0.886883281 0.768455672 0.92670777
0.907405577 0.86379577  0.759258625 0.908502666 0.90579148  0.817439813
0.869440723 0.778185422 0.800988437 0.885927148 0.811211476 0.802932309
0.785080974];%sensitivity

bt=[0.906654853 0.98527289  0.91816009  0.902462055 0.915477538
0.913719876 0.971642453 0.908758043 0.934030489 0.90977892  0.934459462
0.937396994 0.896209577 0.929611542 0.985859583 0.909468618 0.93694572
0.906367741 0.891722103 0.90094522  0.93156408  0.905770115 0.972761166
0.934650746 0.906862242 0.917578657 0.981811892 0.960555418 0.975909378
0.907941657 0.919534438 0.964367434 0.986505691 0.904215257 0.899192437
0.955030441 0.89127477  0.909064212 0.905190465 0.899806225 0.964836142
0.9732597   0.965857206 0.892254905 0.950801841 0.987436101 0.914702487
0.970323846 0.962837446 0.941569896 0.935252889 0.942213234 0.975326054
0.901042224 0.944637104 0.94282935  0.979415014 0.939440625 0.984336508
0.978190054 0.968984315 0.919398871 0.978097185 0.98411116  0.988082341
0.971473421 0.929638624 0.981566707 0.902712755 0.902254802 0.926404132
0.935198232 0.95602367  0.918609615 0.910620992 0.983065473 0.905257303
0.979286096 0.92988011  0.894958239 0.90850487  0.896762726 0.974151238
0.914650274 0.896422526 0.937486111 0.932364963 0.941406866 0.939012372
```

```
0.91097013  0.981140789 0.898929362 0.900527968 0.949253233 0.914888938
0.973041428 0.958736224 0.925715353 0.966138684 0.987483359 0.958895003
0.918057093 0.891750316 0.932049333 0.986324799 0.982544329 0.937989753
0.961941928 0.92095397  0.938774655 0.918541468 0.989795575 0.941646451
0.89766265  0.917726817 0.964171433 0.985582852 0.903957667 0.902163265
0.931611186 0.899762023 0.935003204 0.943984386 0.959889236 0.938170377
0.988183384 0.988132645 0.898435955 0.893955496 0.916125454 0.909072979
0.918635311 0.987031639 0.909438164 0.976258074 0.907865355 0.925333803
0.978966905 0.897238205 0.979359218 0.921552236 0.984621555 0.978638513
0.984093059 0.895391585 0.898494613 0.957411131 0.988563306 0.98491185
0.92861628  0.964018148 0.945909754 0.934066873 0.933560749 0.893793761
0.985435625];%1-specificity

dim=size(at,2);
alpha=0.05;
capB=(dim-1)/alpha-1;

target=zeros(1,dim);%1 if target present
target(1)=1;%1 if target present
mu=at.*(target)+bt.*(1-target);
%-----------------------------------------------------------------------
for z=3:dim
    oa=0;%keep track of errors for error rate calc
    capB=(z-1)/alpha-1;
    z
    a=at(1:z);
    b=bt(1:z);
    for k=1:reps

        c=ones(1,z);%indicators; "1" means that cell is still a
contender
        s=zeros(1,z);%number of detections
        looks=0;%#looks
        stage=0;

        while (sum(c)>1)
            stage=stage+1;
            %         c
            for i=1:z
                if c(i)==1
                    looks=looks+1;
                    r(i)=(rand<=mu(i));
                    s(i)=s(i)+r(i);
                    like(i)=(a(i)^s(i)*(1-a(i))^(stage-
s(i)))/(b(i)^s(i)*(1-b(i))^(stage-s(i)));%likelihood ratios
                end
            end
            maxrat=find(like==max(like));
            for q=1:z
                if c(q)==1
                    ratio(q)=like(maxrat(1))./like(q);%odds ratios
                end
            end
            for j=1:z
```

```
                if ratio(j)>=capB && sum(c)>1;%removes "losers," but
keeps from removing the last contender
                    c(j)=0;
                end
            end

        end

        numLooks(k)=looks;
        declareTgt(k)=find(c==1);

        oa=oa+(declareTgt(k)~=1);

    end

    oarate(z)=oa./reps;

    avgLooks(z)=mean(numLooks);
end
plot(avgLooks)
end
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C

The following tables describe the non-zero columns of the transition matrix used in the adaptive problem of this thesis.

From\To	45	57	58	70	71	76	77	78	88	89	90	100	101	102	103	104	112	113	114
1	0	0	0	0	0	0	0.1	0	0.1	0	0.1	0	0	0.1	0.1	0	0.2	0.2	0.1
2	0	0.1	0	0	0.1	0	0	0	0.1	0	0.1	0	0	0	0	0	0.1	0.4	0.1
3	0	0	0	0	0	0	0.1	0	0	0	0	0.2	0.1	0	0	0	0.2	0	0.4
4	0	0	0	0	0	0.1	0	0	0	0.1	0	0.1	0.1	0.1	0	0	0	0.1	0.4
5	0	0	0	0	0	0	0	0	0	0	0	0.1	0.1	0.1	0	0.1	0	0.2	0.4
6	0	0	0	0	0	0.1	0	0	0.1	0.1	0	0.1	0	0	0.1	0.1	0	0	0.4
7	0	0	0	0.1	0	0	0.1	0	0	0.1	0.1	0.1	0	0.1	0	0.1	0	0	0.3
8	0	0	0	0	0	0	0.1	0	0	0	0	0	0.2	0	0.1	0.1	0.2	0	0.3
9	0	0	0	0	0.1	0	0	0	0	0.1	0.1	0	0.1	0	0	0	0.3	0	0.3
10	0	0	0	0	0	0	0.1	0	0	0.1	0.1	0	0	0.1	0.1	0	0	0.4	0.1
11	0	0	0	0	0	0	0	0	0	0	0.1	0.2	0	0.1	0.1	0	0.2	0.1	0.2
12	0	0	0	0.1	0	0.1	0	0.1	0	0	0	0.1	0	0	0.1	0.1	0.2	0.1	0.1
13	0	0	0	0.1	0	0.1	0	0.1	0	0.1	0.1	0	0	0.1	0.1	0	0.1	0	0.2
14	0	0	0	0	0	0	0	0	0.1	0.1	0.1	0.1	0.1	0	0.1	0	0	0.1	0.3
15	0	0	0	0	0.1	0	0	0	0.1	0.1	0.1	0	0	0	0.2	0.1	0.1	0.1	0.1
16	0	0	0	0	0.1	0.1	0.1	0	0.1	0.1	0	0	0.1	0.1	0	0	0	0	0.3
17	0	0	0	0	0	0.1	0.1	0	0	0	0.1	0	0	0	0	0.3	0.1	0.1	0.2
18	0	0	0.1	0	0	0	0.1	0.1	0	0	0	0	0	0	0.1	0	0.1	0.3	0.2
19	0	0	0	0	0	0	0.1	0	0	0.1	0	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
20	0	0	0	0	0	0	0	0	0.1	0.1	0.1	0	0	0	0.1	0	0	0.3	0.3
21	0	0	0.1	0	0	0	0	0	0	0	0.1	0	0	0.1	0.2	0.1	0.1	0.1	0.2
22	0	0	0.1	0.1	0	0	0	0	0	0	0	0.1	0	0.2	0.1	0	0.1	0.2	0.1
23	0	0.1	0	0	0	0	0	0	0.1	0	0	0	0.1	0	0.1	0.2	0.1	0	0.3
24	0	0	0	0	0	0	0	0	0.1	0.1	0.1	0	0	0	0.1	0	0.1	0	0.5
25	0	0	0	0	0.1	0	0.1	0.1	0	0	0	0.1	0	0	0.1	0.1	0.1	0.1	0.2
26	0	0	0	0	0.1	0	0	0.1	0.1	0.1	0	0.1	0	0	0.1	0.1	0	0.2	0.1
27	0	0	0	0	0	0	0.1	0.1	0	0	0.1	0.1	0	0	0.2	0.1	0	0.2	0.1
28	0	0	0	0	0	0	0	0	0	0	0.1	0	0.2	0.2	0	0.1	0	0.3	0.1
29	0	0	0	0	0	0	0	0	0.1	0	0.1	0	0.1	0.1	0.1	0.2	0	0.1	0.2
30	0	0	0	0.1	0	0	0	0	0	0	0	0	0.2	0.1	0	0	0	0.5	0.1
31	0	0	0	0	0.1	0	0.1	0.1	0	0	0.1	0	0.1	0	0.1	0.1	0	0.1	0.2
32	0	0	0	0	0	0	0.1	0	0.1	0	0	0	0.2	0.1	0	0.2	0	0.1	0.2
33	0	0	0	0	0	0	0	0	0.1	0.1	0.1	0	0.1	0.1	0.1	0	0.1	0	0.3
34	0	0	0	0	0	0	0	0.1	0	0.1	0	0	0	0.2	0	0.2	0.1	0.2	0.1
35	0	0	0.1	0.1	0	0	0.1	0	0.1	0	0	0	0	0	0.2	0.1	0	0.2	0.1
36	0	0	0	0	0	0	0.1	0.1	0.1	0.1	0.1	0	0.1	0.1	0.1	0	0	0.1	0.1
37	0	0.1	0	0	0	0	0	0	0.1	0.1	0	0	0.1	0.1	0	0.2	0.1	0	0.2
38	0	0	0	0.1	0.1	0	0	0	0	0.1	0	0.1	0	0.1	0.1	0.1	0.1	0.1	0.1
39	0	0	0	0	0	0	0	0	0.1	0	0.1	0	0.2	0	0	0	0	0.3	0.3
40	0	0	0	0	0.1	0	0	0	0	0	0.1	0	0.2	0	0.1	0.2	0	0.1	0.2
41	0	0	0	0	0	0	0.1	0	0	0	0	0.1	0	0	0	0.3	0.2	0	0.3
42	0	0.1	0	0	0	0.1	0	0	0	0	0.1	0.1	0.1	0	0.1	0.1	0	0	0.3
43	0	0	0	0.1	0	0.1	0.1	0	0.1	0	0	0	0	0	0.2	0.1	0.1	0.1	0.1
44	0.1	0	0	0	0	0.1	0	0.1	0	0	0.1	0	0.1	0.1	0	0	0.2	0	0.2
45	0	0	0	0	0	0	0	0	0	0.1	0	0	0.2	0.1	0.1	0.1	0.2	0.1	0.1
46	0.1	0	0	0	0	0	0	0.1	0	0.1	0	0.1	0	0.1	0	0	0.1	0.2	0.2
47	0	0	0	0	0	0	0	0.1	0	0	0	0	0.2	0.1	0	0.1	0.1	0.1	0.3
48	0	0	0	0	0	0	0.1	0.1	0	0	0.1	0.1	0	0	0	0	0.3	0	0.3
49	0	0	0	0	0	0	0	0.1	0.1	0	0	0.1	0.1	0.1	0	0.1	0	0	0.4
50	0	0	0	0	0	0.1	0	0	0	0.1	0.1	0.1	0	0.1	0	0.2	0	0.2	0.1
51	0	0	0	0	0	0	0	0.1	0	0	0	0.1	0.1	0	0.1	0.2	0	0.2	0.2
52	0	0	0	0	0	0	0	0	0	0	0.2	0.1	0.1	0	0.1	0.2	0	0.2	0.1

Table 5. TOI transition matrix (non-zero columns only). (Sheet 1 of 3).

From\To	45	57	58	70	71	76	77	78	88	89	90	100	101	102	103	104	112	113	114
53	0	0	0	0	0.1	0	0	0.1	0.1	0.1	0	0.1	0	0.1	0	0	0	0.3	0.1
54	0	0.1	0	0	0	0	0	0.1	0	0	0	0	0	0.1	0.1	0.1	0.1	0	0.4
55	0	0	0	0	0	0	0	0.1	0.1	0.1	0.1	0	0	0	0	0	0.1	0	0.5
56	0	0	0	0	0.1	0	0	0	0.1	0	0	0.1	0	0.1	0.2	0.1	0	0	0.3
57	0	0	0	0	0	0	0	0	0	0.1	0.1	0	0.1	0	0	0	0	0.2	0.5
58	0	0	0	0	0.1	0	0.1	0	0	0.1	0.1	0	0	0.1	0.1	0.1	0.1	0	0.2
59	0	0	0	0	0	0	0	0	0.1	0	0	0.1	0	0	0	0.3	0.2	0.1	0.2
60	0	0	0	0.1	0	0	0	0.1	0	0	0.1	0.1	0	0.1	0.1	0.1	0.1	0	0.2
61	0	0	0	0	0	0.1	0	0.1	0	0.1	0.1	0	0	0.1	0	0	0.2	0	0.3
62	0	0.1	0	0	0	0	0.1	0	0	0.1	0	0.1	0	0	0.1	0	0.3	0	0.2
63	0	0	0	0	0	0.1	0.1	0	0	0.1	0	0	0.1	0	0.2	0	0.1	0	0.3
64	0	0	0.1	0.1	0	0	0	0	0.1	0	0	0.1	0.1	0.1	0	0	0	0	0.4
65	0	0	0	0.1	0	0.1	0.1	0	0.1	0	0	0	0.1	0.1	0.1	0.1	0	0	0.2
66	0	0	0	0.1	0	0	0.1	0	0.1	0	0.1	0	0	0.1	0	0.1	0.2	0	0.2
67	0	0	0.1	0	0	0	0	0	0.1	0.1	0	0	0.1	0.1	0	0.1	0	0.1	0.3
68	0	0	0	0	0.1	0	0.1	0	0.1	0	0.1	0.1	0	0.1	0.1	0	0.1	0	0.2
69	0	0	0	0	0.1	0	0	0	0	0.1	0	0.1	0	0	0.1	0.1	0	0.4	0.1
70	0	0	0	0	0.1	0	0	0	0	0.1	0.1	0	0.1	0.1	0.1	0	0	0.2	0.2
71	0	0	0	0.1	0	0	0.1	0	0	0	0	0.1	0	0	0	0.2	0.1	0.2	0.2
72	0	0	0	0	0.1	0	0	0	0	0.1	0	0.1	0	0.1	0.1	0.2	0	0.1	0.2
73	0	0	0	0.1	0	0.1	0	0.1	0	0	0	0	0	0	0.2	0.2	0.1	0.1	0.1
74	0	0	0	0	0.1	0	0	0	0	0	0.1	0.1	0.1	0.1	0.1	0.1	0	0.2	0.1
75	0	0	0	0	0	0	0	0.1	0.1	0.1	0.1	0	0.1	0	0.1	0	0	0.2	0.2
76	0	0	0	0	0	0	0	0.1	0	0	0	0.1	0.1	0	0	0.3	0.2	0.1	0.1
77	0	0	0	0	0	0	0.1	0	0	0	0	0	0.2	0.1	0	0	0.1	0.3	0.2
78	0	0	0	0	0.1	0	0	0	0	0.1	0	0.1	0.1	0	0.2	0	0	0	0.4
79	0	0	0	0	0.1	0	0.1	0	0.1	0	0	0.1	0	0.1	0.1	0	0	0.2	0.2
80	0	0	0	0.1	0	0.1	0	0.1	0	0.1	0	0.1	0	0.1	0	0	0.2	0.1	0.1
81	0	0	0	0	0	0	0	0.1	0.1	0	0.1	0	0	0	0.1	0.1	0	0.1	0.4
82	0	0	0	0.1	0	0	0	0.1	0	0.1	0	0.1	0	0.1	0.1	0	0	0.2	0.2
83	0	0	0	0	0	0	0.1	0.1	0	0	0	0	0.1	0.2	0.1	0.1	0	0	0.3
84	0	0.1	0	0	0	0.1	0	0	0.1	0	0	0	0.1	0	0.1	0	0.1	0.2	0.2
85	0	0	0	0.1	0.1	0	0.1	0	0	0	0.1	0	0	0.1	0	0.2	0	0	0.3
86	0	0	0	0	0	0.1	0	0.1	0.1	0	0	0.1	0	0.1	0	0	0	0.1	0.4
87	0	0	0	0	0	0	0	0.1	0	0	0	0.1	0.1	0.2	0.1	0	0.2	0	0.2
88	0	0	0.1	0	0	0	0.1	0	0	0	0.1	0	0	0	0	0.2	0.1	0.1	0.3
89	0	0	0	0	0	0	0	0.1	0	0	0.1	0	0.1	0.1	0.2	0.1	0.1	0.1	0.1
90	0	0	0	0	0	0	0	0	0	0	0	0.1	0.1	0.1	0	0.3	0.2	0	0.2
91	0	0	0	0	0	0	0	0.1	0	0	0.1	0	0.1	0	0	0.1	0.3	0	0.3
92	0	0	0.1	0	0	0.1	0	0	0.1	0	0	0	0	0.1	0.1	0	0	0.4	0.1
93	0	0	0	0	0	0.1	0	0	0	0.1	0	0	0	0.2	0.1	0.1	0.1	0.1	0.2
94	0	0	0	0	0	0	0	0	0.1	0	0	0	0	0	0.1	0.1	0.4	0.1	0.2
95	0	0	0.1	0	0	0	0	0	0	0	0.1	0	0.1	0.1	0.2	0	0	0.1	0.3
96	0	0	0	0	0	0.1	0	0	0	0	0	0.1	0	0.2	0.2	0.1	0	0.2	0.1
97	0	0	0.1	0	0	0	0	0	0	0.1	0.1	0.1	0.1	0	0	0.1	0.1	0	0.3
98	0	0	0	0	0	0	0.1	0	0.1	0	0.1	0.1	0.1	0	0.1	0	0.2	0.1	0.1
99	0	0	0	0.1	0.1	0	0	0.1	0	0	0	0	0.1	0	0.1	0.2	0	0	0.3
100	0	0	0	0	0	0.1	0	0	0	0	0.1	0.1	0.1	0	0.1	0	0.3	0	0.2
101	0	0	0	0	0.1	0	0.1	0	0	0	0.1	0	0.1	0	0	0	0.1	0.4	0.1
102	0	0	0	0	0	0.1	0	0	0.1	0.1	0	0	0	0	0.2	0.1	0.1	0.1	0.2
103	0	0	0	0	0	0	0	0	0.1	0	0	0	0.1	0.2	0.1	0.1	0	0	0.4
104	0	0	0	0.1	0	0	0.1	0.1	0	0	0.1	0	0	0	0.1	0.1	0.1	0.2	0.1

Table 6. TOI transition matrix (non-zero columns only). (Sheet 2 of 3).

From\To	45	57	58	70	71	76	77	78	88	89	90	100	101	102	103	104	112	113	114
105	0	0.1	0	0	0	0	0	0	0	0	0.1	0	0.1	0.2	0.1	0.1	0.1	0	0.2
106	0	0	0	0	0	0	0	0	0	0.1	0.1	0.1	0.1	0	0.1	0	0	0.1	0.4
107	0	0	0	0	0.1	0	0	0	0	0.1	0.1	0.1	0.1	0	0.1	0.1	0.1	0.1	0.1
108	0	0	0	0	0	0	0	0.1	0	0.1	0	0.1	0	0.1	0.1	0	0.3	0	0.2
109	0	0	0	0	0	0	0	0	0.1	0.1	0	0.1	0.1	0	0	0.1	0.2	0	0.3
110	0	0.1	0	0	0	0	0	0	0	0	0.1	0.1	0.1	0.1	0	0	0.2	0	0.3
111	0	0	0	0	0	0	0.1	0.1	0	0	0.1	0.1	0	0.1	0.1	0	0	0.3	0.1
112	0	0	0	0	0.1	0	0.1	0.1	0	0	0	0.1	0	0.1	0	0.1	0.1	0.1	0.2
113	0	0	0	0.1	0	0	0	0.1	0	0.1	0	0.1	0.1	0	0.1	0	0.1	0.2	0.1
114	0	0	0.1	0	0	0.1	0	0.1	0	0.1	0	0	0	0.1	0.1	0.1	0.1	0	0.2
115	0.1	0	0	0	0.1	0	0	0	0	0	0	0	0.2	0	0.1	0.2	0.1	0	0.2
116	0	0	0	0	0.1	0.1	0	0.1	0.1	0.1	0	0	0	0	0	0	0	0.1	0.4
117	0	0	0	0	0.1	0.1	0	0	0.1	0	0	0	0.1	0	0.1	0.1	0.1	0.2	0.1
118	0	0	0	0	0	0.1	0	0.1	0.1	0	0.1	0	0	0.1	0.1	0.1	0	0.1	0.2
119	0	0	0	0	0	0	0.1	0	0	0.1	0.1	0	0	0.1	0	0.2	0.1	0	0.3
120	0	0	0	0	0	0	0.1	0	0	0.1	0	0	0	0.2	0	0.1	0	0	0.5
121	0	0.1	0	0	0	0	0	0	0	0.1	0	0.1	0.1	0	0.1	0	0.2	0.2	0.1
122	0	0	0	0.1	0	0.1	0	0	0.1	0.1	0.1	0	0.1	0	0	0.1	0	0	0.3
123	0	0	0	0.1	0	0	0	0	0	0	0.1	0	0.1	0	0.1	0.1	0	0.3	0.2
124	0	0	0	0	0.1	0	0	0	0.1	0	0.1	0.1	0.1	0	0	0	0	0	0.5
125	0	0	0	0	0	0	0	0.1	0	0.1	0.1	0	0.1	0.1	0.1	0.1	0.1	0.1	0.1
126	0	0	0.1	0	0.1	0.1	0	0	0	0	0.1	0	0.1	0	0	0	0.1	0.3	0.1
127	0	0	0	0	0	0.1	0.1	0	0.1	0	0.1	0	0	0.1	0	0	0	0.1	0.4
128	0	0	0.1	0	0	0	0.1	0	0.1	0	0	0	0.1	0.1	0.1	0.1	0.1	0.1	0.1
129	0	0	0	0	0.1	0.1	0.1	0	0	0	0.1	0	0.1	0.1	0	0	0.1	0.2	0.1
130	0	0.1	0	0	0	0.1	0	0	0.1	0	0	0	0.1	0	0.2	0	0.1	0.1	0.2
131	0	0	0	0	0.1	0	0	0.1	0	0.1	0	0.1	0.1	0.1	0.1	0.1	0	0.1	0.1
132	0	0	0	0	0	0.1	0.1	0.1	0	0	0	0	0	0.2	0	0.1	0.1	0	0.3
133	0	0	0	0	0	0.1	0	0	0	0	0.1	0	0	0.1	0.1	0	0.3	0.1	0.2
134	0	0	0	0	0	0	0	0.1	0	0	0	0	0.1	0	0	0.2	0.2	0.3	0.1
135	0	0.1	0	0.1	0	0	0	0	0	0.1	0.1	0	0	0.1	0	0	0.1	0.2	0.2
136	0	0	0	0	0	0	0.1	0	0.1	0	0	0.1	0.1	0	0	0.2	0.2	0.1	0.1
137	0	0	0	0	0	0.1	0	0	0.1	0	0.1	0.1	0	0	0	0.2	0.1	0.2	0.1
138	0	0	0	0.1	0	0	0	0	0	0	0.1	0.1	0	0.2	0.1	0	0.1	0	0.3
139	0	0	0.1	0	0	0.1	0	0.1	0	0.1	0	0	0	0.1	0.1	0.1	0	0.2	0.1
140	0	0	0.1	0	0	0	0	0	0.1	0.1	0	0	0	0.1	0.2	0.1	0	0.2	0.1
141	0	0.1	0	0	0	0.1	0	0	0.1	0	0	0	0.1	0	0	0.2	0.2	0	0.2
142	0	0	0	0	0	0	0	0	0.1	0.1	0.1	0.1	0	0	0.1	0.1	0	0.3	0.1
143	0	0	0	0	0	0	0	0.1	0	0.1	0	0.1	0	0.2	0.1	0	0.1	0	0.3
144	0	0.1	0	0	0.1	0	0	0	0	0	0	0	0.1	0	0.1	0	0.2	0.1	0.3
145	0	0	0	0	0	0.1	0.1	0	0	0.1	0	0.1	0.1	0.1	0	0	0	0.2	0.2
146	0	0	0.1	0	0	0	0	0	0	0	0	0.1	0.1	0	0.2	0.1	0	0	0.4
147	0	0	0	0	0.1	0	0	0.1	0	0	0	0	0.1	0.1	0	0	0.3	0.2	0.1
148	0	0	0	0	0	0	0.1	0	0.1	0.1	0	0	0.1	0.1	0	0	0.2	0	0.3
149	0	0	0	0	0	0	0	0	0	0.1	0.1	0.1	0	0.2	0	0.2	0.1	0.1	0.1
150	0	0	0.1	0	0	0	0	0.1	0	0	0.1	0	0	0	0	0.1	0.1	0.1	0.4
151	0	0.1	0	0	0	0	0	0.1	0	0.1	0	0	0	0.1	0	0	0	0.4	0.2
152	0	0	0	0.1	0	0.1	0	0	0	0	0.1	0	0.1	0	0.1	0.1	0	0.2	0.2
153	0	0	0	0	0	0	0	0	0	0	0	0.1	0	0.1	0.1	0.3	0	0.2	0.2
154	0	0	0	0	0	0	0	0	0	0.1	0	0.1	0.1	0.1	0.1	0.1	0.2	0	0.2
155	0	0	0	0.1	0	0	0	0.1	0	0	0.1	0.1	0	0	0	0.2	0.2	0.1	0.1
156	0	0	0	0.1	0	0	0	0	0.1	0	0.1	0	0	0	0.2	0	0.1	0	0.4

Table 7. TOI transition matrix (non-zero columns only). (Sheet 3 of 3).

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

Benkoski, S., Monticino, M.G., and Weisinger, J.R., 1991. "A Survey of the Search Theory Literature," *Naval Research Logistics*, vol. 38, pp. 469–494.

Brown, S.S., 1977. "Optimal and Near Optimal Search for a Target with Multiple Scenario Markovian, Constrained Markovian, or Geometric Memory Motion in Discrete Time and Space," Daniel H. Wagner, Associates, Memorandum Report.

Calhoun, G.L., Draper, M.H., Nelson, J.T., and Ruff, H.A., 2007. "Advanced Display Concepts for UAV Sensor Operations: Landmark Cues and Picture-In-Picture," *Human Factors and Ergonomics Society Annual Meeting Proceedings, Aerospace Systems*, pp. 121-125, Human Factors and Engineering Society.

Carl, R.G., 2003. *Search Theory and U-Boats in the Bay of Biscay*, Master's Thesis, Air Force Institute of Technology, Wright-Patterson Air Base, OH.

Dambreville, F. and Le Cadre, J.P., 2001. "Distribution of Continuous Search Effort for the Detection of a Target with Optimal Moving Strategy," *Signal and Data Processing of Small Targets 2001, Proceedings-SPIE The International Society for Optical Engineering*, Conference 13, O.E. Drummond, ed., vol. 4473, pp. 175–185.

Dambreville, F. and Le Cadre, J.P., 1999. "Detection of a Markovian Target with Optimization of the Search Efforts under Generalized Linear Constraints," *Theme 3, Institut National de Recherche en Informatique et en Automatique, No. 3815*.

DelBalzo, D.R., and Hemsteter, K.P., 2002. "GRASP Multi-sensor Search Tactics against Evading Targets," Oceans *'02 MTS/IEEE*, vol. 1, pp. 54–59.

Dell, R.F., Eagle, J.N., Martins, G.H.A., and Santos, A.G., 1996. "Using Multiple Searchers in Constrained-Path, Moving-Target Search Problems," *Naval Research Logistics*, vol. 43, pp. 463–480.

Dembo, A. and Zeitouni, O., 1993. *Large Deviations Techniques and Applications*, Boston: Jones and Bartlett.

Dobbie, J., 1974. "A Two-Cell Model of Search for a Moving Target," *Operations Research*, vol. 22, pp. 79-92.

Dodd, T. and Apopei, B., 2007. "Development of a Mini-UAV for Urban Environments," [http://www.shef.ac.uk/content/1/c6/05/45/20/IET%20MAV%20talk.ppt], accessed 17 November 2008.

Hellman, O., 1972. "On the Optimal Search for a Randomly Moving Target," *SIAM Journal of Applied Math*, vol. 22, pp. 545–552.

Iida, K., 1972. “Ido Mokuhyobutsu no Tansaku (The Optimal Distribution of Searching Effort for a Moving Target),” *Keiei Kagaku (Japan),* vol. 16, pp. 204–215.

Jones, J.S., 2006. *Modeling Detection Strategies to Battle Improvised Explosive Devices,* Master’s Thesis, Naval Postgraduate School, Monterey, CA.

Kadane, J.B., 1971. “Optimal Whereabouts Search,” *Operations Research,* vol. 19, pp. 894–904.

Kiefer, J. and Wolfowitz, J., 1952. “Stochastic Estimation of the Maximum of a Regression Function,” *Annals of Mathematical Statistics*, vol. 23, no. 3, pp. 462–466.

Koopman, B.O., 1980. *Search and Screening: General Principles with Historical Applications,* Oxford: Pergamon Press.

Kress, M., Szechtman, R., and Jones, J.S., 2008. “Efficient Employment of Non-Reactive Sensors,” *Military Operations Research*, vol. 13, pp.45–57.

Kushner, H.J. and Yin, G.G., 2003. *Stochastic Approximation and Recursive Algorithms and Applications*, 2nd ed., New York: Springer.

Lions, J.L., 1971. *Optimal Control of Systems Governed by Partial Differential Equations,* pp. 248-249, New York: Springer-Verlag.

Lohr, W.J., 1992. *Modeling the Search for a Randomly Moving Target by a Patrolling Searcher,* Master’s Thesis, Naval Postgraduate School, Monterey, CA.

Malone, G.J., 2004. *Ranking and Selection Procedures for Bernoulli and Multinomial Data,* Doctoral Thesis, Georgia Institute of Technology.

Moser, H.D., 1990. *Scheduling and Routing Tactical Aerial Reconnaissance Vehicles,* Master’s Thesis, Naval Postgraduate School, Monterey, CA.

Owen, P., Martin, R., and Carriger, T., 2005. “Shadowing IEDs,” *Unmanned Systems*, vol. 2, pp.13-16.

Peot, M.A., and others, 2005. “Planning Sensing Actions for UAVs in Urban Domains,” [http://markpeot.googlepages.com/SPIEUAV26Aug2005--Final.pdf], accessed 17 November 2008.

Pollock, S.M., 1970. “A Simple Model of Search for a Moving Target,” *Operations Research,* vol. 18, pp. 883–903.

Pursiheimo, U., 1976. “A Control Theory Approach in the Theory of Search when the Motion of the Target is Conditionally Deterministic with Stochastic Parameters,” *Applied Mathematics and Optimization*, vol. 2, pp. 259–264.

Reber, D.N., 2007. *Optimized Routing of Unmanned Aerial Vehicles for the Interdiction of Improvised Explosive Devices,* Master's Thesis, Naval Postgraduate School, Monterey, CA.

Riese, S.R., 2006. "Templating an Adaptive Threat: Spatial Forecasting in Operations Enduring Freedom and Iraqi Freedom," *Engineer*, pp. 42–43.

Riese, S.R., 2008. "Threat Maps for Baghdad CFD Trials," presentation to the Johns Hopkins University Applied Physics Laboratory.

Robbins, H. and Monro, S., 1951. "A Stochastic Approximation Method," *Annals of Mathematical Statistics,* vol. 23, no. 3, pp. 400–407.

Ross, S. M., 1996. *Stochastic Processes,* 2nd ed., New York: Wiley & Sons Inc.

Saretsalo, L., 1973. "On the Optimal Search for a Target Whose Motion is a Markov Process," *Journal of Applied Probability,* vol. 10, pp. 747–756.

Sato, H. and Royset, J.O., 2008. "Path Optimization for the Resource-constrained Searcher," [http://faculty.nps.edu/joroyset/docs/SatoRoyset.pdf], accessed 30 July 2009.

Scheidt, D.H., 2008. "Mission-level Autonomy for Unmanned Vehicle Teams," [http://grasp.upenn.edu/~chpeng/icra08workshop/DavidScheidt.pdf], accessed 17 November 2008.

Siegmund, D., 1985. *Sequential Analysis: Tests and Confidence Intervals,* New York: Springer-Verlag.

Spall, J.C., 2003. *Introduction to Stochastic Search and Optimization,* New York: Wiley & Sons, Inc.

Stewart, T. J., 1979. "Search for a Moving Target when Searcher Motion is Restricted," *Computers & Operations Research*, vol. 6, no. 3, pp. 129–140.

Stone, L.D., 1977. "Search for Targets with Generalized Conditionally Deterministic Motion," *SIAM Journal of Applied Mathematics,* vol. 33, pp. 456–468.

Stone, L.D. and Kadane, J.B., 1981. "Optimal Whereabouts Search for a Moving Target," *Operations Research,* vol. 29, pp. 1154–1166.

Tierney, L. and Kadane, J.B., 1983. "Surveillance Search for a Moving Target," *Operations Research,* vol. 31, pp.720–738.

Tognetti, K.P., 1968. "An Optimal Strategy for Whereabouts Search," *Operations Research,* vol. 16, pp. 209–211.

Wagner, D.H., Mylander, C., and Sanders, T.J., 1999. *Naval Operations Analysis,* Annapolis, MD: U.S. Naval Institute.

Washburn, A. R., 1983. "Search for a Moving Target: The FAB Algorithm," *Operations Research*, vol. 31, no. 4, pp. 739-751.

Washburn, A. R., 1989. *Search and Detection,* 2nd ed., pp. 2-15, Hanover, MD: Operations Research Society of America.

Weinstein, A.L. and Schumacher, C., 2007. "UAV Scheduling via the Vehicle Routing Problem with Time Windows (Preprint)," Air Vehicles Directorate, Air Force Materiel Command, Air Force Research Laboratory, Wright-Patterson Air Force Base, OH.

Wieland, J.R. and Nelson, B.L., 2004. "An Odds-Ratio Indifference Zone Selection Procedure for Bernoulli Populations," *Proceedings of the 2004 Winter Simulation Conference, Department of Industrial Engineering & Management Science,* Northwestern University, Evanston IL.

Yan, I. and Blankenship, G.L., 1987. "Search for Randomly Moving Targets I: Estimation," *Proceedings of the 26th Conference on Decision and Control, Systems Research Center and Electrical Engineering Department*, University of Maryland, College Park, MD.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Mr. John F. Keane
   The Johns Hopkins University Applied Physics Laboratory
   Laurel, Maryland

4. Assistant Professor Roberto Szechtman
   Naval Postgraduate School
   Monterey, California

5. Assistant Professor Johannes O. Royset
   Naval Postgraduate School
   Monterey, California